Manuelli

FLODI

Governo FDU per sistemi con CPU19 e A7 Descrizione di funzionamento

olivetti

Codice 3956180 C (1)
(661.30.1)

FLODI - Governo FDU per sistemi con CPU 19 e A7

Descrizione di Funzionamento

Codice pubblicazione : 3956180 C

Codice aggiornamento : 1

1980

INDICE

INTRODUZIONE . . . . . Pag. 1

SINTESI DELLE OPERAZIONI SVOLTE . . . . . " 2

SCHEMA A BLOCCHI GENERALE . . . . . " 3

PRINCIPIO DI FUNZIONAMENTO DEL FLODI . . . . . " 4

- Avvio dei comandi di Classe C o B . . . . . " 4
- Flusso riassuntivo delle interruzioni a Liv. 3B . . . . . " 6
- Posizionamento del carrello per le STIO di classe C e seek . " 7
- Posizionamento del carrello nel caso di "Home" . . . . . " 8
- Flusso riassuntivo delle operazioni svolte durante le interruzioni di posizionamento del carrellino . . . . . " 9
- Controllo del tempo d'attesa : "Conta tempo" . . . . . " 10
- Flusso funzionale per le interruzioni di conta tempo . . . . " 11
- Controllo del numero dei settori da trattare e del numero dei giri fatti dal disco . . . . . " 12
- Flusso delle operazioni da svolgere durante le interruzioni a livello 3A per funzioni di governo dopo la conta tempo o per fine operazione di classe C . . . . . " 14
- Ricerca del settore da trattare . . . . . " 15
- Formato traccia per Floppy Disk . . . . . " 16
- Registrazione di settori "Deleted" o "Undeleted" . . . . . " 16
- Flusso della logica di registrazione . . . . . " 18
- Funzionamento della lettura . . . . . " 19
- Flusso della logica di lettura . . . . . " 20
- Funzionamento della scansione . . . . . " 21
- Flusso della logica della scansione . . . . . " 23

TABELLE RIASSUNTIVE . . . . . " 24

DESCRIZIONE DELLA LOGICA DI FUNZIONAMENTO . . . Pag. 27

• Logica della selezione e della risposta di selezione . . . " 27
• Logica dell'interruzione per risposta a comando . . . . . " 29
• Logica di posizionamento del carrello . . . . . " 29
• Logica di conta tempo . . . . . " 30
• Logica di conta degli index . . . . . " 31
• Logica della ricerca dell'identificatore . . . . . " 31
• Logica della lettura . . . . . " 35
• Logica della registrazione . . . . . " 37
• Logica della verifica . . . . . " 39
• Logica della scansione . . . . . " 40

NOTE DI PRINCIPIO SUL PROGRAMMA DI IMMATRICOLAZIONE . . . . . " 43

• Logica dell'operazione "Registra traccia" . . . . . " 44
• Logica di PONI in locale . . . . . " 48
• Logica di cambia disco . . . . . " 48
• Tabelle da ritagliare . . . . . " 49

SCHEMI A BLOCCHI . . . . . " 51

• Tabella riassuntiva delle risposte sui BUS . . . . . " 53
• Fase di selezione . . . . . " 54
• Fase di selezione più comando . . . . . " 55
• Fase di ricerca traccia (SEEK) . . . . . " 56
• Fase di ricerca settore (Confronto identificatore) . . . . " 59
• Lettura campo dati . . . . . " 62
• Registrazione dati . . . . . " 65
• Comando di verifica . . . . . " 68
• Comando di scansione . . . . . " 70
• Logica di fine . . . . . " 72
• Comando di "Poni in locale" . . . . . " 73
• Comando di "Cambia disco" . . . . . " 73

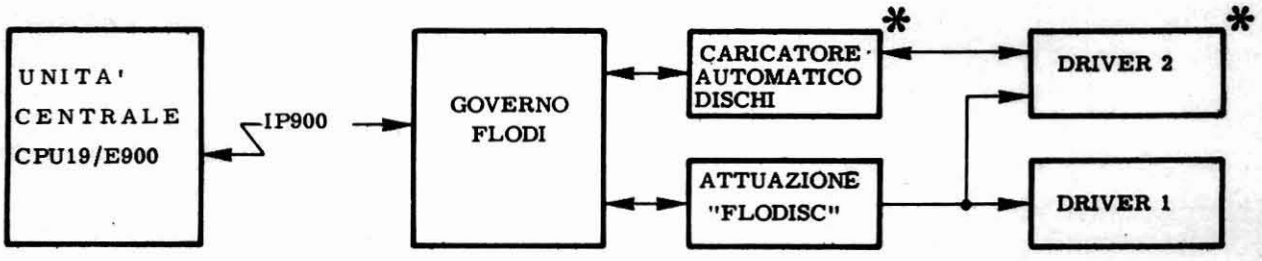
INTRODUZIONE

Il governo FLODI permette la connessione di un FDU, con 1 o 2 driver, ai sistemi che utilizzano le unità centrali CPU 19 o E 900.

Il governo FLODI é costituito da 2 piastre standard per micrologici.

Le modalità d'interfaccia tra unità centrale e il governo sono quelle definite dall'interfaccia standard IP 900.

La connessione logica del sistema realizzabile é riportata in figura 1.



* Unità opzionali

FIG. 1

Come risulta dalla figura 1, il FLODI può anche pilotare l'eventuale caricatore automatico di dischi.

Per quanto concerne la sistemazione fisica del FLODI nell'ambito dei singoli sistemi si rimanda alla documentazione specifica di sistema.

N. B. E' costruita una seconda versione del governo FLODI che prende il nome di FLOD2.
I e differenze fra le due versioni sono limitate al circuito stampato e ed altre piccole cose per cui la presente publicazione è valida per entrambi i modelli: le piccole differenze sono evidenziate sugli schemi logici.

SINTESI DELLE OPERAZIONI SVOLTE

Le operazioni che interessano il FLODI possono essere riassunte nella tabella 1.

Tab. 1

OPERAZIONE STI0 o SI0	CODICE SOFTWARE	CLASSE COMANDO
PONI DRIVER IN STATO DI LOCALE	4.0	B
CAMBIA DISCO	4.1	B
POSIZIONAMENTO SU UNA TRACCIA QUALSIASI (SEEK)	4.4	B
POSIZIONAMENTO SU TRACCIA "00" (HOME)	4.8	B
POSIZIONAMENTO E LETTURA DI UNO O PIU' SETTORI	C.0	C
POSIZIONAMENTO E REGISTRAZIONE DI UNO O PIU' SETTORI	8.1	C
POSIZIONAMENTO E REGISTRAZIONE DI UNO O PIU' BLOCCHI "DELETED"	8.0	C
POSIZIONAMENTO E VERIFICA DI UNO O PIU' BLOCCHI	8.2	C
POSIZIONAMENTO E SCANSIONE SU $\geqslant$ (SCAN$\geqslant$)	8.6	C
POSIZIONAMENTO E SCANSIONE SU = (SCAN=)	8.4	C
POSIZIONAMENTO E SCANSIONE SU $\leqslant$ (SCAN$\leqslant$)	8.5	C
POSIZIONAMENTO E REGISTRA TRACCIA	8.3	C
POSIZIONAMENTO E REGISTRA TRACCIA DIFETTOSA	8.8	C

Le STI0 o SI0 di PONI MOD0 128 e PONI MOD0 256 fanno parte del set di operazioni specifiche del sottosistema con FLODI ed FDU ma vengono svolte nell'ambito della sola U.C.

Per ulteriori informazioni sul modo di utlizzazione dei comandi descritti si colsulti la documentazione specifica di sistema.

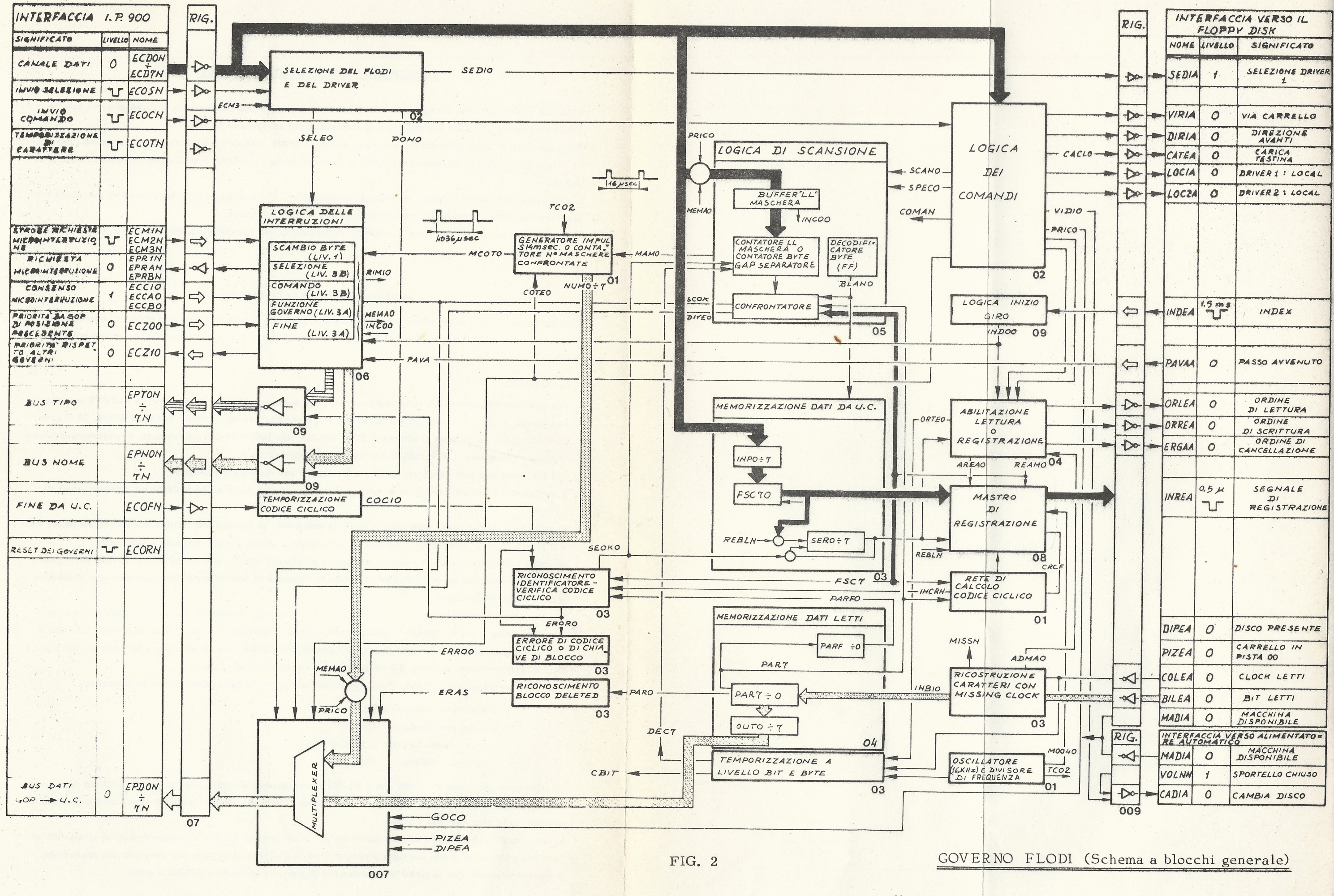


FIG. 2

GOVERNO FLODI (Schema a blocchi generale)

PRINCIPIO DI FUNZIONAMENTO DEL FLODI

Prima di addentrarci nel dettaglio del funzionamento, illustreremo con l'ausilio di flussogrammi e dello schema a blocchi di figura 2 il principio di funzionamento del governo FLODI.
Richiameremo il funzionamento del colloquio standard IP 900 laddove esso interessa più da vicino l'hardware del FLODI al fine di dividere nettamente le funzioni svolte dal FLODI e quelle svolte da U. C. durante il colloquio.
Consigliamo inoltre chi ci legge, di seguire lo schema a blocchi generale e di individuare ed eventualmente annotare sui DISL i segnali che nomineremo nella nostra spiegazione.
Successivamente, dopo aver visto il principio di funzionamento spiegheremo nel dettaglio il FLODI dal punto di vista puramente hardware.

- AVVIO DEI COMANDI DI CLASSE C o B -

Ogni volta che l'U. C. vuole avviare un comando di classe B o C deve verificare lo stato della periferica.
Per poter richiedere il bus di stato al FLODI, l'U. C. seleziona il FLODI eseguendo una microistruzione E SE.

- 1ª selezione

Con la microistruzione ESE viene emesso il nome hardware (tabella 2) del governo FLODI e del trascinatore che si vuole interessare sul bus ECD0 ÷ 7, accompagnato dagli strobe ECOS ed ECOT. Se il governo é collegato, su di esso si setterà il F. F. SELE0, che indica FLODI selezionato, e contemporaneamente viene predisposto il F. F. SEDI0 che seleziona il driver 1 o 2.
SELE0 causa una richiesta interruzione a livello 3B per risposta a selezione.
La selezione é inviata a livello 4 di microprogramma, come viene anche indicato nel flussogramma di figura 3.
Si noti che a livello base, dopo aver avviato la selezione, si innesca una routine di conta di 300 µsec. che é il tempo entro il quale l'U. C. aspetta l'interruzione a livello 3B per risposta a selezione.
Se entro i 300 µsec. non arrivasse la risposta a selezione, l'U. C. considererebbe il FLODI scollegato e lo segnalerebbe. Nel caso di FLODI collegato la richiesta di interruzione dovrebbe arrivare immediatamente dopo la selezione.
Quando arriva la richiesta di interruzione e l'U. C. l' accetta emettendo ECCB, il FLODI predispone i bus tipo, nome e stato. Durante l'interruzione, l'U. C. intruduce i tre bus e dopo aver verificato che si tratta di una risposta a selezione (bus tipo) da parte del FLODI (bus nome) azzera la conta dei 300 µsec. e tronca l'interruzione a livello 3B eseguendo una microistruzione COM0.
Una volta tornata a livello base, l'U. C. esamina lo stato ed in particolare rileva le condizioni:

- macchina disponibile (MADI)
- disco non presente (DICA)
- trascinatore in stato di locale (LOCA)
- governo occupato (GOC0)

Per governo occupato s'intende che il FLODI sta eseguendo un'operazione di posizionamento del carrello, di lettura o registrazione del disco, o di cambio disco avviata con una STI0 precedente a quella che ha causato la selezione del FLODI.
Si noti, infatti, che il F. F. SELE0 sta in set per i soli periodi relativi alle interruzioni di livello 3B per risposta a selezione e non per tutto il tempo in cui il FLODI é impegnato per eseguire una operazione.
E' perciò possibile che ci siano la segnalazione di governo occupato e SELEO a zero.

L'U. C., se non rileva impedimenti all'esecuzione della STI0 e restando a livello base, prepara i parametri relativi all'operazione da eseguire. S'intende con questo, che l'U. C. predispone il numero di passi di cui deve spostarsi il carrellino, gli indirizzi per lo scambio dati e così via.
L'esame del bus di stato é fatto a livello 4 e non a livello 3B al fine di ridurre la durata dell'interruzione e quindi di ridurre l'occupazione del canale.

- 2ª selezione

Dopo aver predisposto i parametri dell'operazione, l'U. C. emette una nuova selezione del governo FLODI che provoca una nuova richiesta di interruzione per risposta a selezione; l'U. C. introduce i bus nome e tipo, si accerta che l'interruzione arrivi dal FLODI e che si tratta di una risposta a selezione, quindi esegue due microistruzioni CAE per l'emissione di due comandi. Ogni microistruzione CAE emette sul bus dati un byte (comando) accompagnato dagli strobe ECOC ed ECOT.
Il primo comando é temporaneamente staticizzato su un buffer a 8 bit e viene sostituito sullo stesso buffer dal secondo comando.
Il comando emesso con le CAE varia in funzione della operazione che si deve eseguire. Il comando posiziona i segnali LOC1 o LOC2 e LOCA nel caso di STI0 per il posizionamento in LOCALE o avvia il carrellino nel caso di HOME, SEEK o altre STI0 di classe C.
Si noti come a questo punto tutte le STI0 possono appena dirsi avviate (tranne il caso di "poni in locale" che già si é esaurita).
L'U. C., dopo aver emesso i comandi, tronca l'interruzione per risposta a selezione con una microistruzione COM0. La stessa COM0 resetta SELE0 e predispone in set il F. F. COMA0.

- Interruzione per risposta a comando

Il F. F. COMA0 fa sì che sul FLODI nasca una richiesta d'interruzione per risposta a comando anch'essa a livello 3B. Quando l'interruzione é accettata dall'U. C. vengono di nuovo emessi dal FLODI i bus nome, tipo e stato. L'U. C. introduce i tre bus e con una COM0 annulla l'interruzione per risposta a comando.
Una volta tornata a livello base, l'U. C. esamina lo stato del governo e vede se il governo si dichiara in LOCALE o OCCUPATO da cui può trarre le conclusioni per sapere se il governo ha accettato il comando stesso.
Se il governo ha accettato il comando, l'U. C. inizia una nuova fase di programma e considera la STI0 già finita.
Facciamo notare che nel caso di STI0 : SEEK, HOME o STI0 di classe C, l'esecuzione dell'operazione é appena avviata. Va di per sé che a questo punto é il governo FLODI che col suo hardware segue l'FDU e richiede di sua iniziativa l'intervento dell'U. C. interrompendola a livello 3A o 3B o 1.
Se nello svolgimento del programma l'U. C. incontrasse una nuova STI0 rivolta all'FDU, essa provvederebbe ad eseguire una nuova selezione da cui ricava l'informazione di unità occupata finché la STI0 precedente non é terminata.

FLUSSO RIASSUNTIVO DELLE INTERRUZIONI A LIVELLO 3B

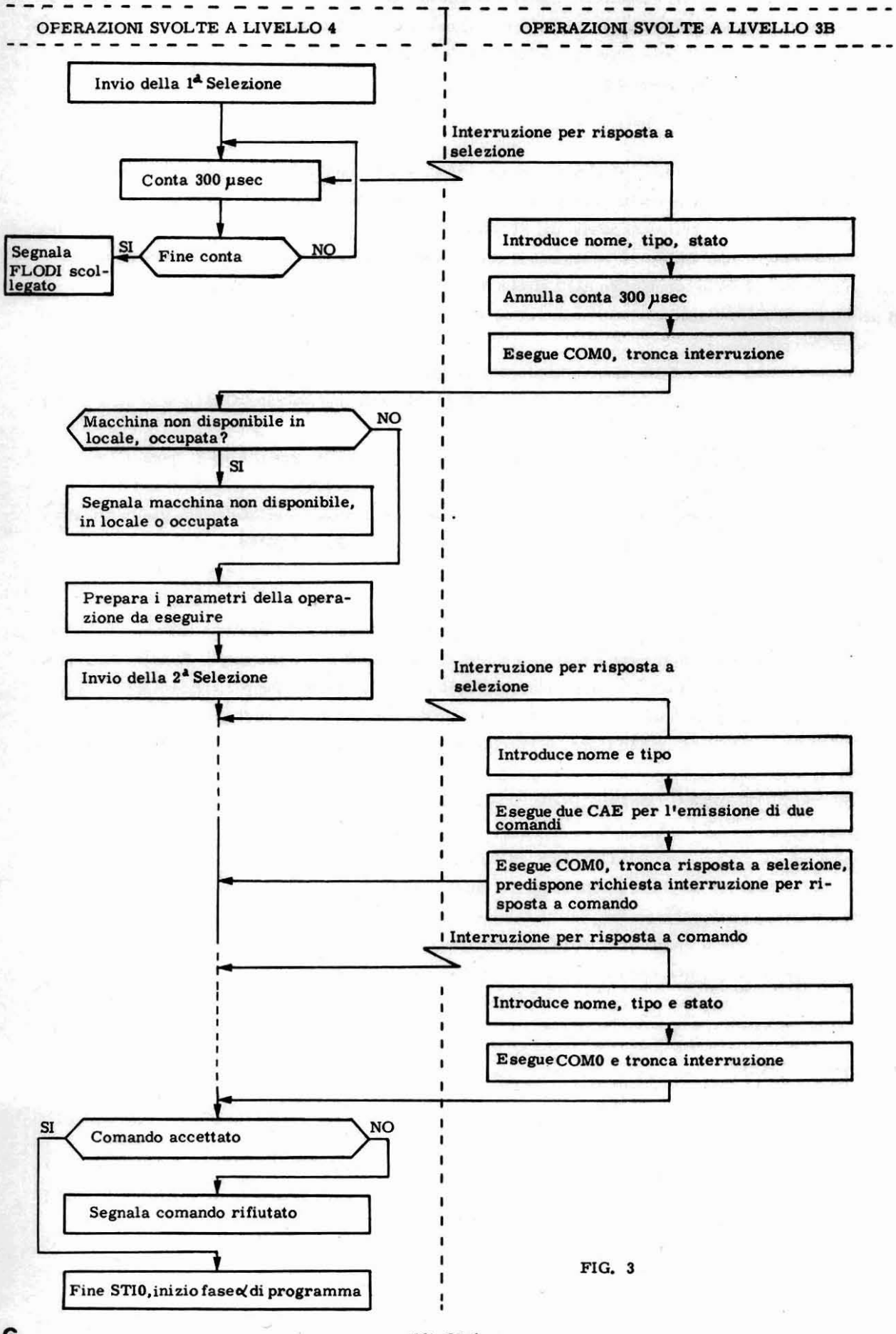


FIG. 3

- POSIZIONAMENTO DEL CARRELLO PER LE STI0 DI CLASSE C e SEEK -

Il posizionamento del carrello viene sempre predisposto con le due microistruzioni CAE emesse durante la seconda selezione nel caso di STI0 HOME, SEEK o STI0 di classe C.
Il via al posizionamento é dato predisponendo DIRI0 già con la prima CAE e posizionando VIRI0 con la seconda.
Facciamo notare che le CAE sono due solo perché é richiesto che DIRI0 sia già posizionato nel momento in cui nasce VIRI0 per un buon funzionamento della piastra FLODISC, altrimenti con uno stesso comando potrebbero essere posizionati DIRI0 e VIRI0.
Dopo aver posizionato VIRI0, il carrello comincia a mouversi e per ogni traccia su cui si sposta,la piastra di attuazione FLODISC emette un impulso PAVA. Al fine di ridurre l'hardware del FLODI, il controllo del posizionamento é fatto con l'aiuto della U. C.
Il governo infatti si limita ad interrompere l'U. C. a livello 3A ad ogni impulso PAVA che riceve. Nel frattempo l'U. C. che, preparando i parametri dell'operazione, aveva caricato in un contatore ricavato in RAM il numero di passi di cui deve spostarsi la testina, ogni volta che accetta l'interruzione a livello 3 A deconta di 1 il contatore. Quando, durante l'interruzione di livello 3A, l'U. C. rileva che il suo contatore é azzerato il posizionamento può considerarsi finito.
Durante le interruzioni di livello 3A causate da PAVA, l'U. C. si limita ad introdurre i bus nome, tipo e stato e a decontare il contatore dei passi.
Al solito, l'interruzione termina con una COM0 eseguita da U. C. E' solo durante l'interruzione in cui il contatore va a zero che l'U. C. emette un nuovo comando eseguendo una CAE. Il byte emesso in questa circostanza richiama VIRI0 a zero per fermare il carrellino e posiziona i F. F. CLABO e CATE0.
CATE0 avvia sul FLODI un tempo di attesa di 40 msec durante i quali il carrellino deve posizionarsi.
CLAB0 controlla invece il caricamento della testina nel caso si tratti di comandi di classe C.
Nel caso di operazione SEEK non viene comandato il caricamento della testina ma viene comunque avviato il tempo d'attesa durante il quale il carrellino deve stabilizzarsi e durante il quale il FLODI continua sempre a generare la condizione di governo occupato.

- Considerazioni sul termine "interruzioni per funzioni di governo"

Le interruzioni che il FLODI esegue ad ogni PAVA, sarebbero facilmente evitabili se l'U. C. emettesse sul FLODI il numero di passi di spostamento e quindi il FLODI stesso decontasse i passi avvenuti.
Si é scelta la strada di far fare su U. C. il conteggio dei PAVAA solo per ridurre l'hardware del FLODI stesso. Poiché nel caso di posizionamento del carrello, l'U. C. viene interrotta per fare delle funzioni che potrebbero essere svolte dal governo; queste interruzioni a livello 3A vengono chiamate interruzioni per "funzioni di governo".
Il posizionamento del carrello non é il solo caso in cui il FLODI interrompe l'U. C. a livello 3A per funzioni di governo,come avremo modo di constatare di seguito.

- Caso particolare di posizionamento con carrello già in posizione corretta

L'operazione svolta varia a seconda che si esegua una SEEK, nel qual caso la STI0 può già dirsi eseguita o una STI0 di classe C in cui comunque occorre caricare la testina per leggere o registrare il disco.
Nel caso di SEEK, l'operazione termina dopo la prima selezione e non s'inizia nemmeno la seconda selezione. Nel caso di STI0 di classe C, durante la 2ª selezione, il comando emesso, anziché posizionare DIRI0 e VIRI0, posiziona CATE0 e CLAB0 come avviene normalmente alla fine del posizionamento.
L'operazione inizia quindi con il caricamento della testina e un tempo d'attesa durante il quale la testina si stabilizza.

- POSIZIONAMENTO DEL CARRELLO NEL CASO DI HOME -

Una STI0 "HOME" inizia come tutte le altre STI0 con una doppia selezione. Sulla seconda selezione viene emesso il comando di via carrello.
Durante le interruzioni per "funzioni di governo" richieste ad ogni PAVA, l'U. C. introduce come al solito il bus stato ed esamina se il carrellino é giunto in pista "00" (PIZE).
Quando il carrellino é in pista "00", l'U. C. emette il comando di arrestare il carrello e avvia con CATE0 il tempo d'attesa di 40 msec entro i quali il carrellino deve stabilizzarsi.

-Caso di "HOME" con carrellino già in pista "00"

Il comando di via carrello é dato come al solito durante la seconda selezione. Si ricorda però che, predisponendo VIRI0 ad 1 e DIRI0 a zero, sull'attuazione FLODISC viene impedito il movimento del carrellino mentre sono comunque abilitati i segnali PAVAA.
Al primo PAVAA generato si interrompe l'U. C. per "funzione di governo" e l'U. C., come al solito, nel caso di HOME, introduce nel bus di stato l'informazione di carrellino in pista "00".

FLUSSO RIASSUNTIVO DELLE OPERAZIONI SVOLTE DURANTE LE INTERRUZIONI DI POSIZIONAMENTO DEL CARRELLINO

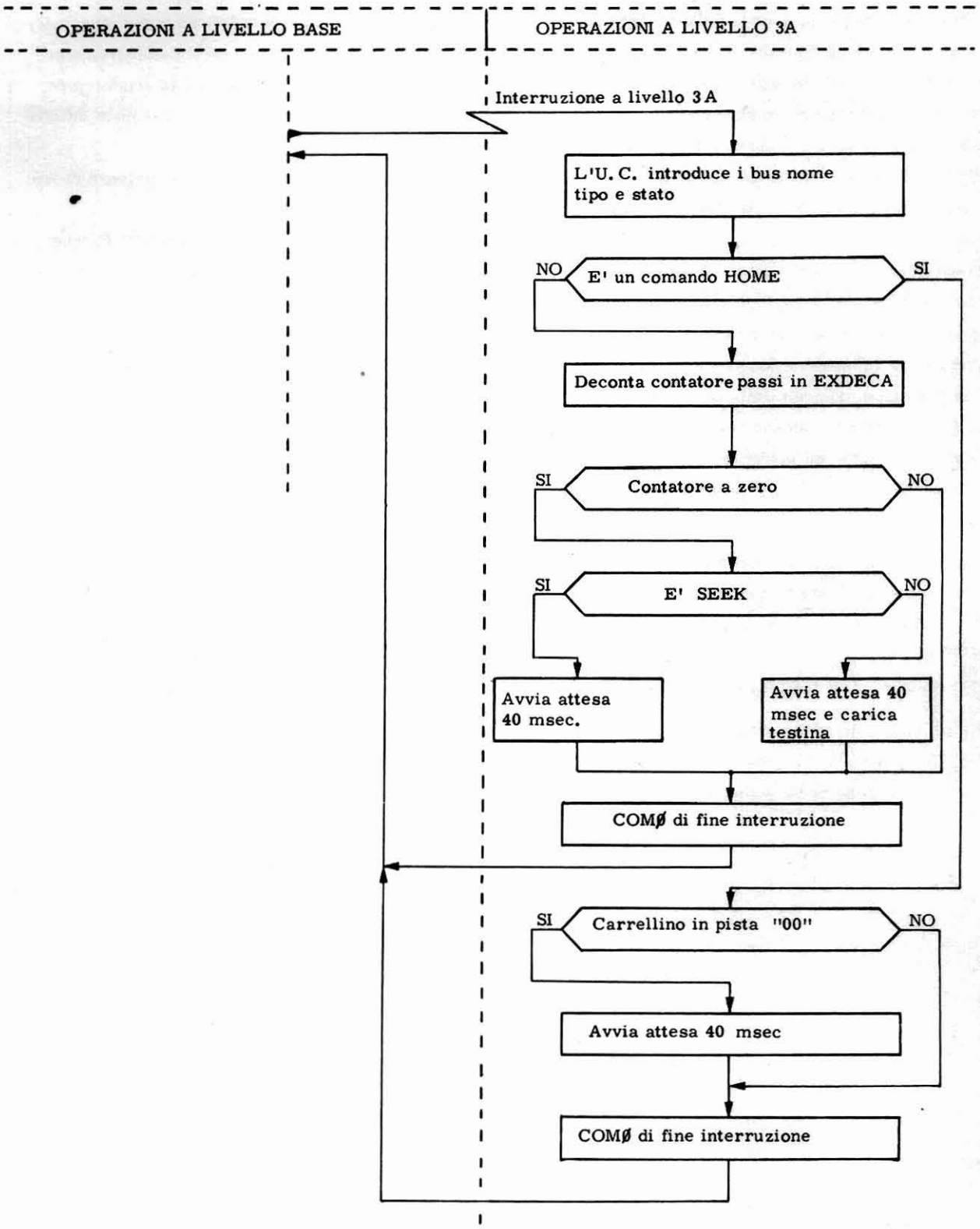


FIG. 4

- CONTROLLO DEL TEMPO D'ATTESA : "CONTA TEMPO" -

Ogni qualvolta termina un posizionamento del carrello o si avvia una operazione di lettura e registrazione, si avvia un tempo d'attesa di circa 40msec durante i quali il carrellino deve stabilizzarsi definitivamente e la testina di lettura deve essere convenientemente caricata. Il segnale CATE0 controlla la conta tempo. Sfruttando dell'hardware necessario anche in altre funzioni il FLODI si limita a generare il segnale MC0T0 su cui si succede un treno di impulsi con periodo di 4096 μsec.

MC0T0 é ottenuto per divisione della frequenza di TC02 il quale a sua volta é ottenuto da un divisore di frequenza comandato da un oscillatore a 16 MHz.

Anziché inserire un contatore decimale che conti dieci impulsi MC0T0, si ricorre all'aiuto dell'U. C. che viene interrotta "per funzioni di governo" ad ogni MC0T0. L'U. C., nel momento in cui avvia il tempo d'attesa, predispone la configurazione 10 in un contatore dell'EXDECA e ad ogni interruzione per "funzione di governo" deconta di uno il contatore.

Durante la 10ª interruzione per "funzione di governo" il contatore si azzera e l'U. C. esegue due CAE per l'emissione di due comandi. La prima CAE emette un byte che viene staticizzato sul buffer dei comandi. Il primo comando emesso setta anche il F. F. PRIC0 per cui il byte emesso con la seconda CAE non é abilitato ad entrare sul buffer dei comandi ma entra sul buffer di lunghezza della chiave come si può vedere sullo schema a blocchi. Il comando emesso con la prima CAE varia in funzione della STI0 che viene eseguita ed in particolare:

a) azzera il buffer e annulla la conta tempo con "Home" e "Seek"

b) annulla la conta tempo, conferma il caricamento della testina e posiziona i comandi SCAN0, REG00 e SPEC0 con una qualsiasi STI0 di classe "C".
   I tre segnali SPEC0 SCAN0 e REG00 assumono configurazioni diverse in funzione della STI0 che viene eseguita e controllano su FLODI le operazioni di lettura, verifica, scansione o registrazione.

La seconda CAE viene sempre eseguita anche se in realtà ha significato solo nel caso di STI0 del tipo "SCAN" in cui l'U. C. emette sul "buffer lunghezza chiave" il complemento a 256 della lunghezza della chiave di scansione al quale l'U. C. somma già un 1.

FLUSSO FUNZIONALE PER LE INTERRUZIONI DI CONTA TEMPO

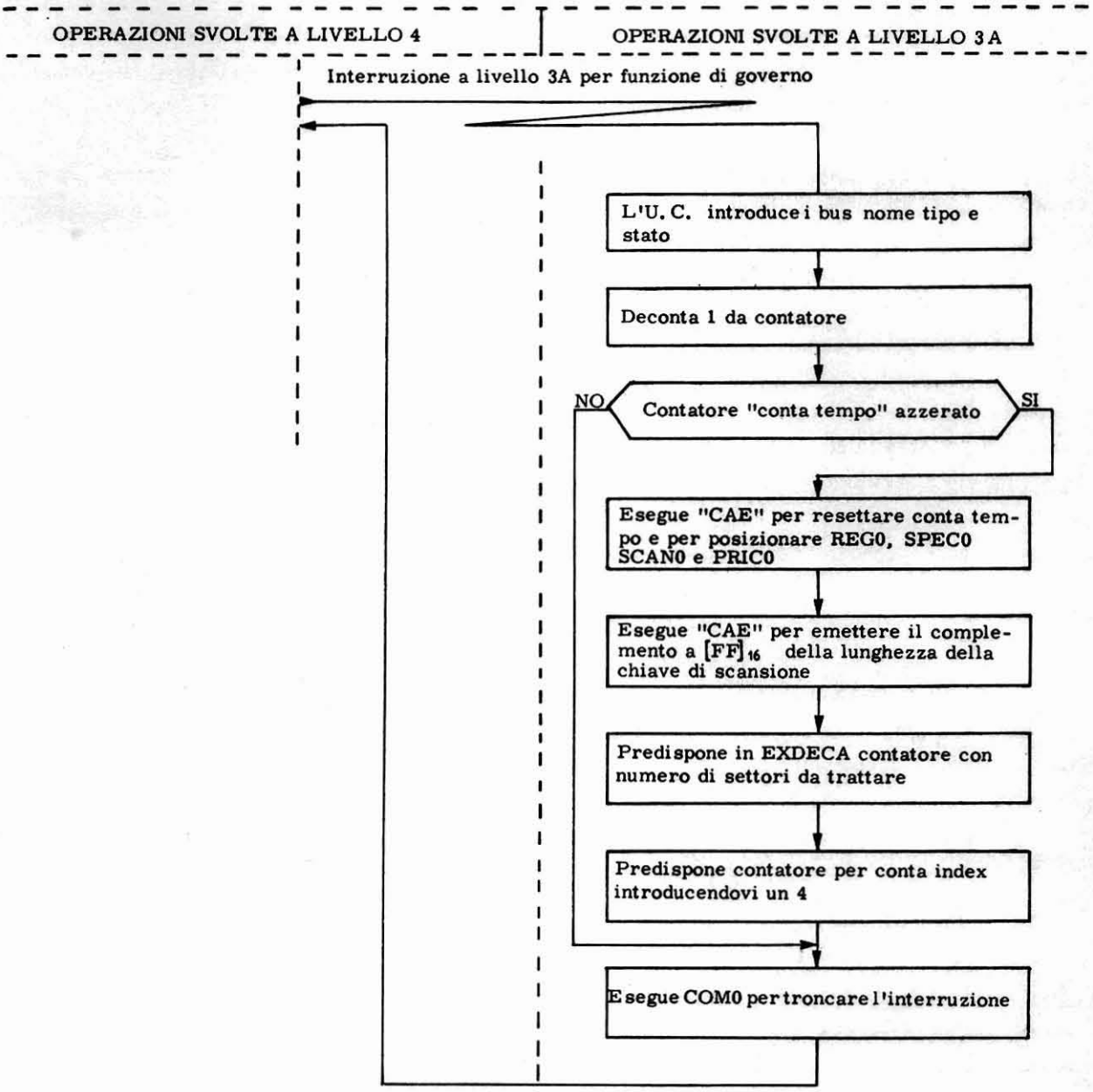


FIG. 5

- CONTROLLO DEL NUMERO DEI SETTORI DA TRATTARE E DEL NUMERO DEI GIRI FATTI DAL DISCO -

Dopo aver atteso i 40 msec per la stabilizzazione del carrellino e il caricamento della testina, inizia l'esecuzione della lettura, registrazione, verifica o scansione.

- Interruzioni per conta settori da trattare

Trascuriamo per un momento l'operazione di "registra traccia" e pensiamo a tutte le altre STI0 di classe C. Esse possono trattare un numero variabile di settori contenuti in una unica traccia. Il numero di settori é memorizzato nella U. C. ed esattamente nell'EXDECA come anche l'indirizzo del primo byte da trattare.

L'operazione di lettura, verifica, registrazione o scansione inizia comunque con una lettura del disco al fine di individuare l'inizio del primo settore. Nel capitolo seguente vedremo in linea di principio come viene riconosciuto l'identificatore di un settore. Limitiamoci per ora a dire che, una volta individuato l'identificatore del settore voluto, mentre si legge la chiave di blocco "KB"(nel caso di lettura, verifica o scansione)o mentre la si registra (registrazione), il FLODI interrompe a livello 3 A per "funzione di governo" l'U. C.

L'U. C. dopo aver accettato l'interruzione introduce i bus nome, tipo e stato. L'U. C. esamina lo stato, sottrae 1 dal contatore dei settori da trattare, aggiorna l'indirizzo del prossimo settore da trattare nell'ambito dell'EXDECA e tronca l'interruzione con una microistruzione COM0.

Alla fine del settore,il FLODI,di sua iniziativa,si predispone di nuovo comunque alla lettura per individuare l'identificatore del secondo settore da trattare e quindi, sulla chiave di blocco del secondo settore, emetterà una seconda interruzione per "funzione di governo".

Quando, durante una interruzione per funzione di governo per la conta dei settori da trattare, il contatore va a zero, l'U. C. segnala al FLODI che quello che si sta per trattare é l'ultimo settore eseguendo due CAE che emettono gli stessi byte già emessi durante l'ultima "funzione di governo" di conta tempo.

I due byte sono smistati l'uno sul buffer dei comandi e l'altro sul buffer lunghezza chiave e confermano su entrambi i buffer il loro contenuto. I due comandi sono emessi sfruttando il solito microprogramma della conta tempo solo per avere il segnale INC00.

- Interruzioni per fine operazione di classe "C"

Il segnale INC00 se non si é in conta tempo, memorizza sul FLODI che il settore che si sta trattando é l'ultimo di quelli da trattare. Alla fine del settore, il FLODI provvederà ad interrompere quindi di sua iniziativa l'U. C. a livello 3 A. Questa volta il FLODI indica sul bus "tipo" che si tratta di una interruzione per "fine operazione di classe "C". Durante questa interruzione, l' U. C. preleva i bus nome, tipo e stato ed esaminando lo stato può verificare se ci sono stati errori in lettura rilevati col controllo del codice ciclico, se tra i settori letti c'erano dei settori "deleted" e nel caso di scansione può rilevare se la scansione é stata verificata.

Durante l'interruzione, l'U. C. esegue una microistruzione CAE con la quale emette un comando che resetta il buffer dei comandi facendo così "scaricare" le testine e richiamando a zero i F. F. SCAN0, REG0 e SPEC0 che controllavano su FLODI il tipo d'operazione da eseguire.

Il comando emesso setta PRIC0 e sul bus dati EPD0 + 7 viene abilitato non più il bus di stato bensì il contenuto del buffer NUM0 + 7, come si può vedere sullo schema a blocchi. Dopo aver eseguito la CAE, l'U. C. esegue sempre una microistruzione per introdurre il bus dati del quale tiene conto solo nel caso di"SCAN".

Sul buffer NUM0 + 7 durante la scansione viene memorizzato il numero di chiave sulla quale si é verificata la scansione. Dopo aver introdotto per la 2^4 volta il bus EPD0 + 7,l'U. C. tronca l'interruzione.

Facciamo notare che solo alla fine delle operazioni di classe C nasce l'interruzione per fine, mentre nel caso di operazioni non di classe C, la STI0 termina senza una interruzione per fine.

- Controllo del numero dei giri del disco

L'operazione di classe C deve terminare entro un massimo di tre giri dalla fine dei 40msec attesi per la stabilizzazione del carrellino.

Per questo durante la decima interruzione per "funzioni di governo" di conta tempo l'U. C. oltre a predisporre il contatore col numero dei settori da trattare predispone un secondo contatore con un 4 per contare i giri del disco.

Il FLODI infatti interrompe l'U. C. a livello 3A per "funzioni di governo" ad ogni "index" rilevato dall'FDU. L'U. C. distingue le interruzioni a livello 3A per funzioni di governo di conta index da quelle di conta dei settori esaminando il bus di stato. Se l'operazione STI0 non termina prima che il contatore degli index venga azzerato, l'U. C., sulla quarta interruzione di conta index, di sua iniziativa, considera finita l'operazione e provvede a resettare il buffer dei comandi emettendo un byte $[00]_{16}$ con una CAE. In questo caso l'operazione di classe C termina senza una interruzione per fine.

Fine "SCAN" data da FLODI

Abbiamo visto come l'interruzione di fine é normalmente data dall'U. C., che conteggia i settori trattati. Nel caso però che la scansione venga verificata, il FLODI, di sua iniziativa, al termine del settore su cui si é verificata la scansione provvede ad interrompere l'U. C. a livello 3A per "fine operazione di classe C". In particolare, se la scansione é verificata sull'ultimo settore da trattare, l'interruzione per fine é avviata contemporaneamente per due vie, una quella di scansione verificata e l'altra di ultimo settore da trattare.

FLUSSO DELLE OPERAZIONI DA SVOLGERE DURANTE LE INTERRUZIONI A LIVELLO 3A PER FUNZIONI DI GOVERNO DOPO LA CONTA TEMPO O PER FINE OPERAZIONE DI CLASSE C.

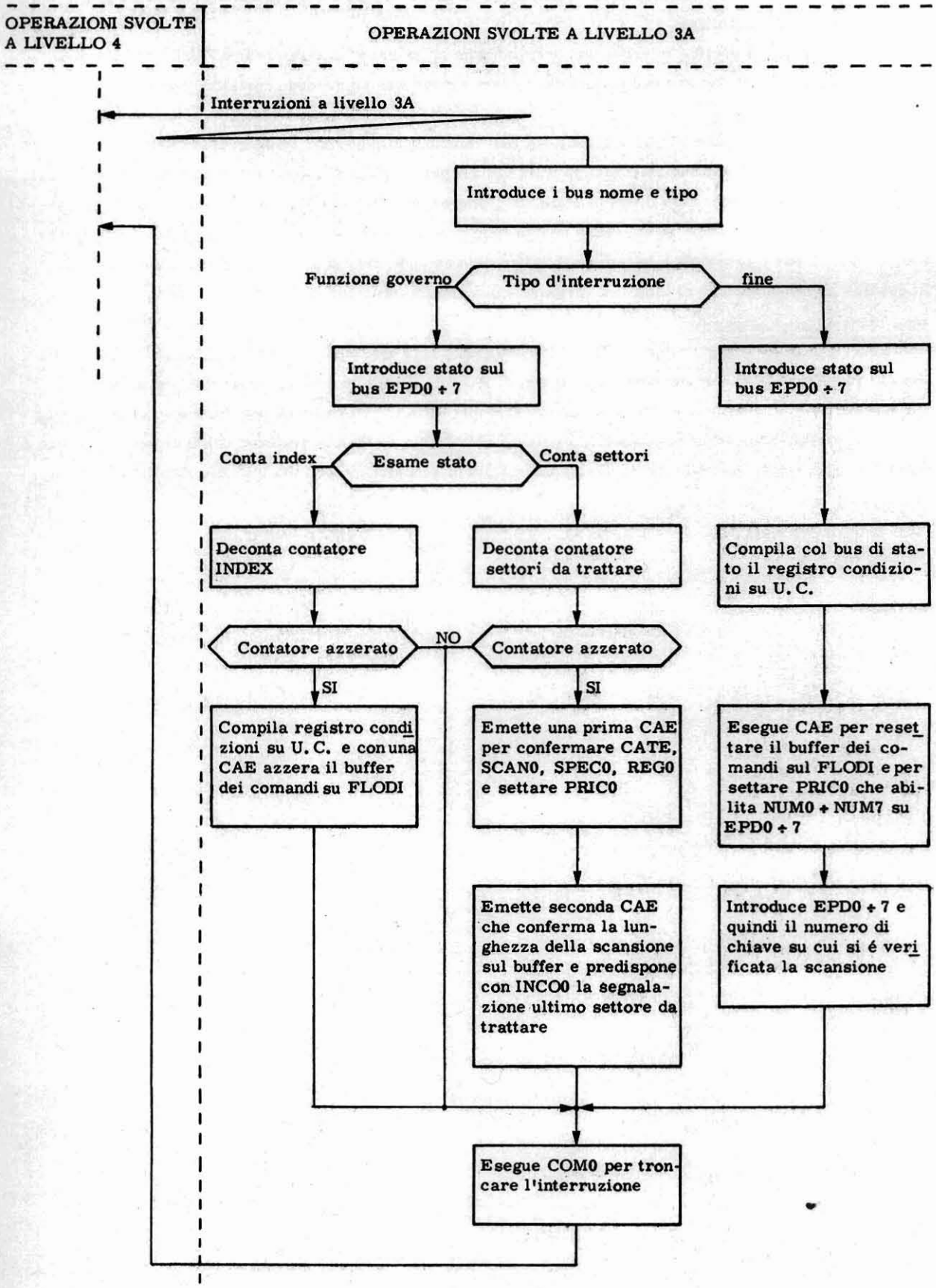


FIG. 6

- RICERCA DEL SETTORE DA TRATTARE -

In lettura, verifica, scansione o registrazione, per individuare il settore su cui si vuole operare si esaminano tutti gli identificatori registrati sulla traccia fino a trovare quello voluto. Per comodità riportiamo in figura 8 lo schema del formato della traccia.

Ogni identificatore è preceduto da una chiave di settore "K S" con una configurazione di clock "CKS" e una configurazione bit "BKS" che possiamo esprimere rispettivamente come $[C7]_{16}$ e $[FE]_{16}$.

Sul FLODI c'è una rete di hardware in grado di individuare un byte con missing clock, la quale in corrispondenza di una chiave di settore e di una chiave di blocco emette i segnali MISS e ADMA0 come risulta dallo schema a blocchi. Sul FLODI, il segnale MISS innesca una serie di interruzioni a livello 1 in corrispondenza di ogni byte letto, e l'U. C. emette, per ogni interruzione, nell'ordine i byte "BKS", [TRACCIA] $[00]_{16}$ [SETTORE] $[00]_{16}$ più altri due byte che potranno servire nella registrazione ma che non vengono presi in esame durante il riconoscimento dell'identificatore.

Nella temporizzazione di figura 7 sono riportati nella giusta relazione di fase i segnali letti, il segnale MISS, le richieste di interruzioni a livello 1 (EPRI0) e i byte emessi da U. C. sul buffer INP0 ÷ 7.

Il byte è emesso da U. C. in parallelo ed esce da INP7 in serie dopo di che shifta sempre in serie sugli "shift register" FSC0 ÷ 7 e SER0 ÷ 7 sui quali subisce sempre un ritardo di due byte come è indicato sullo schema a blocchi.

I byte letti sul disco vengono shiftati sui buffer PAR0 ÷ 7 e PARO8-F0 ognuno dei quali introduce un ritardo di un byte. Come risulta dalla temporizzazione da PARF0 e da FSC70 escono dei byte omogenei che possono essere confrontati bit a bit, in maniera seriale su una rete hardware di riconoscimento dell'identificatore. Su tale rete vengono confrontati i byte "BKS", TRACCIA $[00]_{16}$ "Settore" $[00]_{16}$ e se tutti i bit sono uguali il segnale EROR0 rimane a zero altrimenti è messo ad 1.

Le interruzioni a livello 1 sono avviate sul FLODI da MISS, ma il loro numero dipende dall' U. C. che emette ECOF0 e determina un segnale COCI0 di controllo del codice ciclico e che arresta la sequenza di interruzioni a livello 1. ECOF0, COCI0 e la fine delle interruzioni sono indicate sul diagramma di figura 7. Durante la lettura, i byte letti, presi all'uscita di PAR7 sono portati in una rete di calcolo del codice ciclico realizzata in hardware sul FLODI.

L'uscita di tale rete INCRN deve essere costantemente a zero durante il tempo COCI0 altrimenti comunque viene posizionato il F. F. EROR0.

EROR0 si troverà in set alla fine di COCI0 se il confronto dei primi 4 byte dell'identificatore più il byte " BKS" è stato positivo e se il controllo del codice ciclico non ha segnalato errori. Se EROR0 è a zero alla caduta di COCI0 significa che è trovato il giusto identificatore e quindi viene generato SEOK (Sector O. K.).

TEMPORIZZAZIONE
Ricerca Identificatore

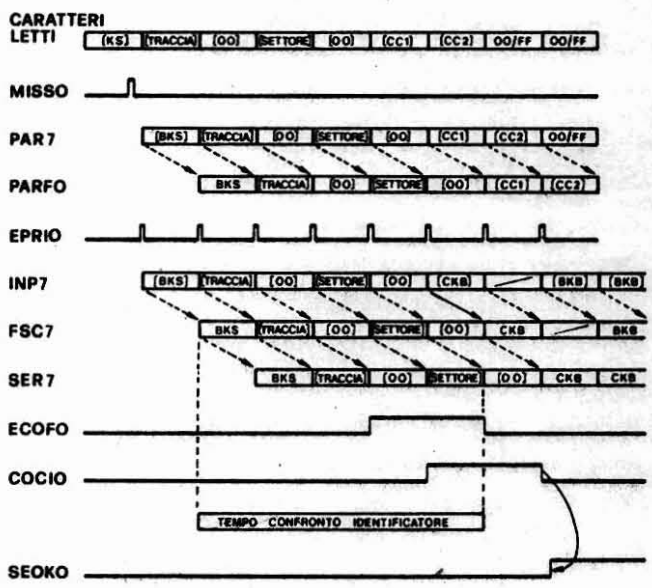
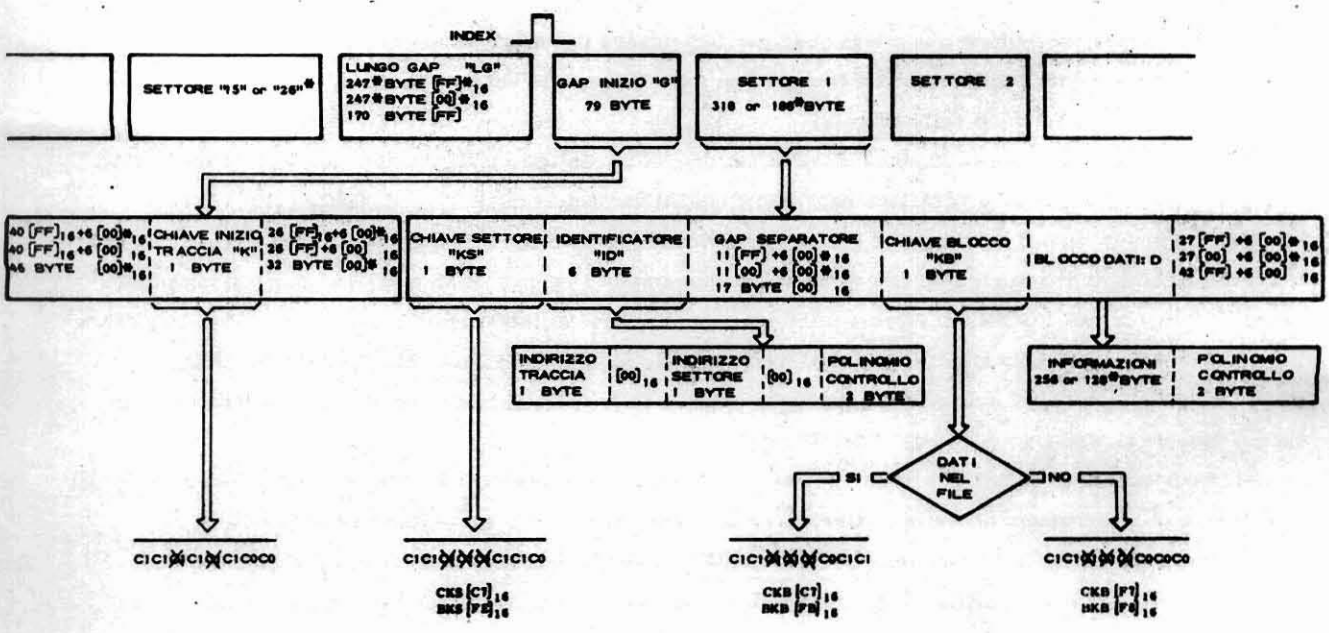


FIG. 7

FORMATO TRACCIA PER FLOPPY DISK



* I numeri di byte contrassegnati da un asterisco sono relativi al formato con 128 byte per settore.
- I clock barrati non sono fisicamente registrati nel disco, sono normalmente chiamati "clock mancanti".
- La traccia "00" è per entrambi i formati organizzata su blocchi a 128 BYTE.

FIG. 8

- REGISTRAZIONE DI SETTORI "DELETED" O "UNDELETED"

La registrazione su uno o più settori inizia con una doppia selezione, un posizionamento del carrellino se questo è necessario, una conta tempo o la ricerca del primo settore da registrare. Una volta individuato il settore da registrare si lasciano scorrere i primi 11 byte che sono contati dal FLODI dopo di che si fa nascere il segnale ORRE e si dà inizio alla registrazione come è anche indicato in figura 9.

Dati registrati

Nell'ambito del settore vengono registrati gli ultimi sei byte del gap separatore, la chiave di blocco, e i dati comprendenti le informazioni vere e proprie e due byte di codice ciclico.

Dopo aver registrato il codice ciclico si richiama il segnale ORRE e quindi si arrestano le registrazioni e ci si predispone a ricercare un eventuale settore successivo.

Precisiamo che il gap separatore conserva nei primi undici byte ciò che era stato registrato all'immatricolazione (byte $[FF]_{16}$ o $[00]_{16}$.

In realtà, i primi due byte registrati possono avere contenuti incerti perchè sebbene la logica del FLODI invii byte $[00]_{16}$, i transitori che seguono l'attivazione del circuito di scrittura possono causare la registrazione di informazioni incoerenti.

Il FLODI controlla la registrazione del gap separatore senza richiedere l'aiuto dell'U. C. e provvede a registrare quindi la chiave di blocco KB.

Per registrare la chiave di blocco occorre sopprimere la registrazione di alcuni clock e registrare come bit la configurazione bit BKB $[F8]_{16}$ o $[FB]_{16}$ a seconda che il blocco sia "deleted" o "undeleted".

Sia la configurazione dei clock CKB che la configurazione bit BKB della chiave di blocco sono emessi da U. C. durante la ricerca settore come risulta anche dalla figura 7.
Le due configurazioni sono conservate rispettivamente su SER0 ÷ 7 (CKB) e su FSC0 ÷ 7 (BKB) e alla fine del gap separatore controllano la registrazione della chiave di blocco.

- Meccanismo delle interruzioni

Mentre si registra la chiave di blocco, il FLODI avvia una richiesta d'interruzione a livello 3A per la conta dei settori trattati e di una serie di interruzioni a livello 1 per chiedere le informazioni da registrare all'U. C.
Le interruzioni a livello 1 sono effettuate una per ogni byte da registrare e durante ogni singola interruzione l'U. C. emette un byte.
Il byte è memorizzato su INP0 ÷ 7 e quindi attraverso FSC0 ÷ 7 è trasferito sul disco. L'unità centrale controlla i byte da registrare e con l'ultimo byte emette il segnale ECOF0 che causa la nascita di COCI0 che controlla la registrazione di due byte di codice ciclico calcolati sul FLODI.
La fine del segnale COCI0 causa anche la fine delle richieste interruzioni a livello 1.
Con la caduta di COCI0 se durante l'ultima interruzione a livello 3A per conta settori è nata la segnalazione di ultimo settore si avvia una richiesta interruzione per fine sempre a livello 3A.

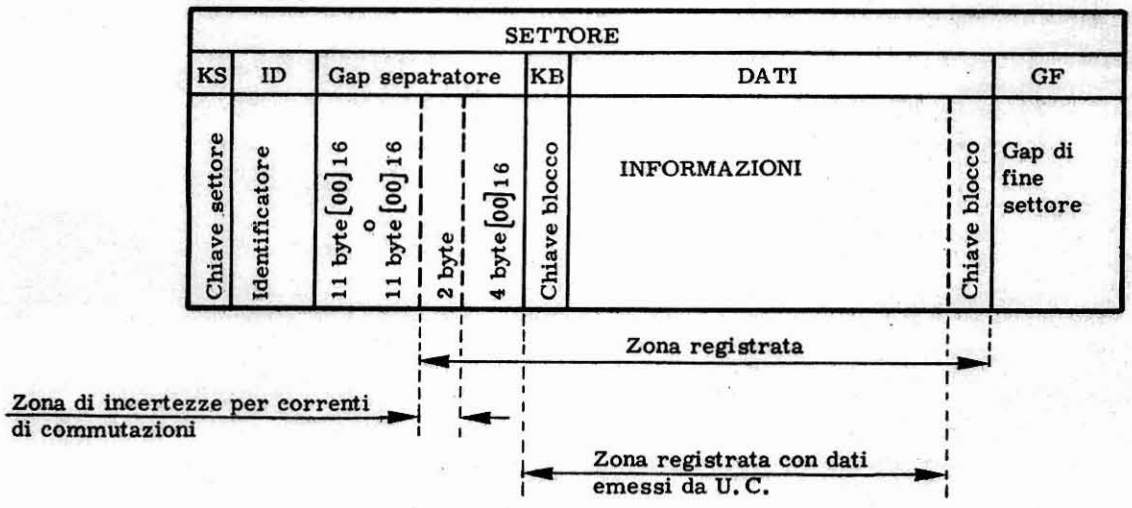


FIG. 9

FLUSSO DELLA LOGICA DI REGISTRAZIONE

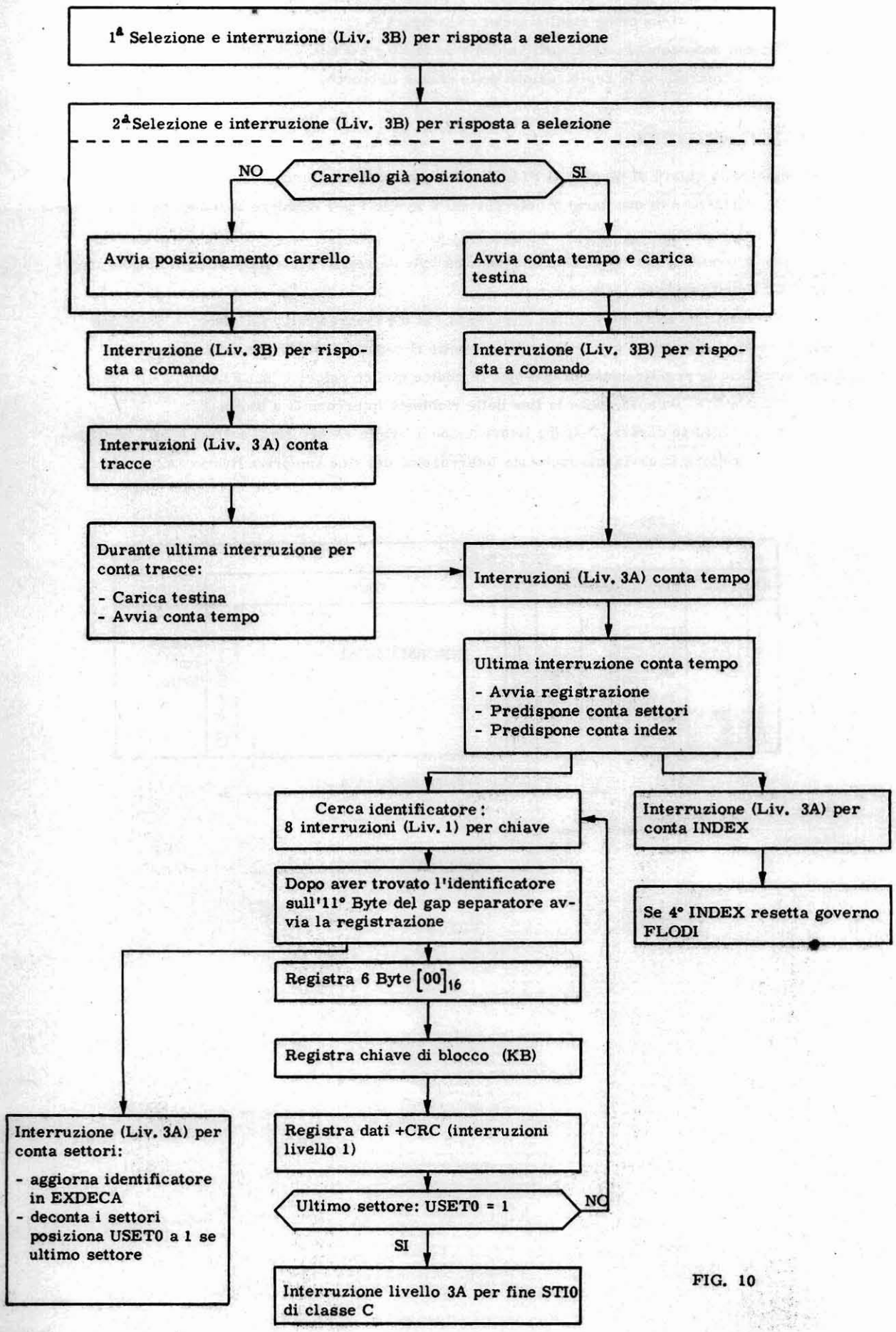


FIG. 10

661.30.1

- FUNZIONAMENTO DELLA LETTURA -

La lettura di un settore inizia come al solito con una ricerca di un identificatore. Una volta riconosciuto l'identificatore, si inizia la lettura.

Si ricorda però che in corrispondenza del 11° e 12° byte del gap separatore, sono registrate delle informazioni incoerenti per cui in corrispondenza di essi si disabilita la lettura facendo cadere il segnale ORLE0 (fig. 11). ORLE0 sta basso per 64 µsec e quindi si riabilita un riaggancio della temporizzazione sull'attuazione FLODISC.

Durante gli ultimi 4 byte del gap separatore il P. L. O; dell'FDU viene sincronizzato e quando viene letta la chiave di blocco (carattere con missing clock) si risincronizza anche la temporizzazione di carattere sul FLODI.

A partire dalla chiave di blocco inizia una serie di interruzioni a livello 1 durante le quali i byte letti sono inviati all'U. C. dopo essere passati attraverso PAR0 ÷ 7 e OUT0 ÷ 7.

Durante la lettura della chiave di blocco si esegue anche una interruzione a livello 3A per la conta dei settori letti.

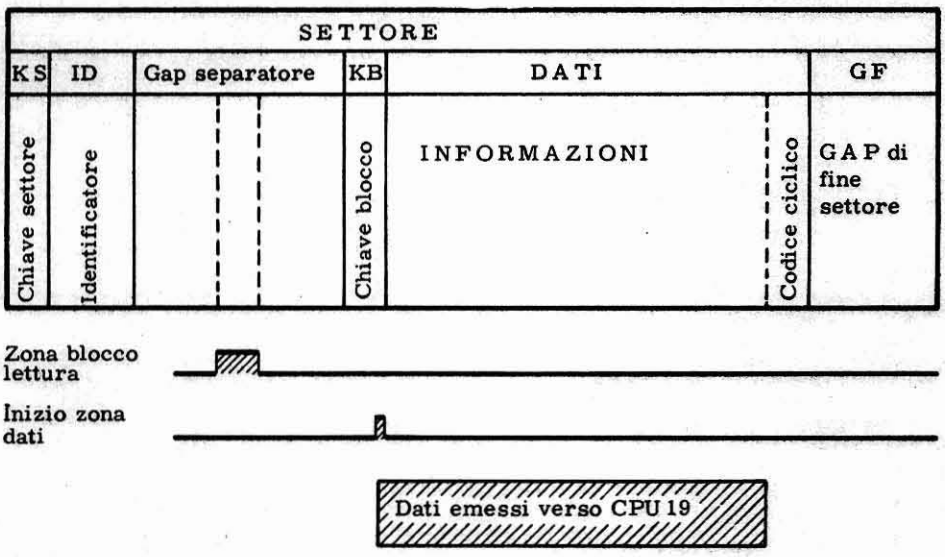


FIG. 11

- Controlli sulla chiave di blocco "KB"

Dopo aver letto la chiave di blocco, il FLODI controlla i tre bit meno significativi della chiave per stabilire se si stratti veramente di una chiave di blocco e quindi per vedere se il blocco che si sta leggendo è "deleted" o no.

In caso di blocco "deleted", il FLODI posiziona ad 1 il segnale ERAS0 e l'informazione relativa è inviata all'U. C. durante l'interruzione per "fine".

- Controlli sulla lettura

La lettura termina quando l'U. C. emette un segnale ECOF0 che come al solito abilita la nascita di COCI0 durante il quale il codice ciclico letto sul disco viene comparato a quello che durante la lettura è stato calcolato sul FLODI.

In caso di errore viene posizionato EROR0 e quindi ERR0 che durante l'interruzione per fine è inviato all'U. C.

FLUSSO DELLA LOGICA DI LETTURA

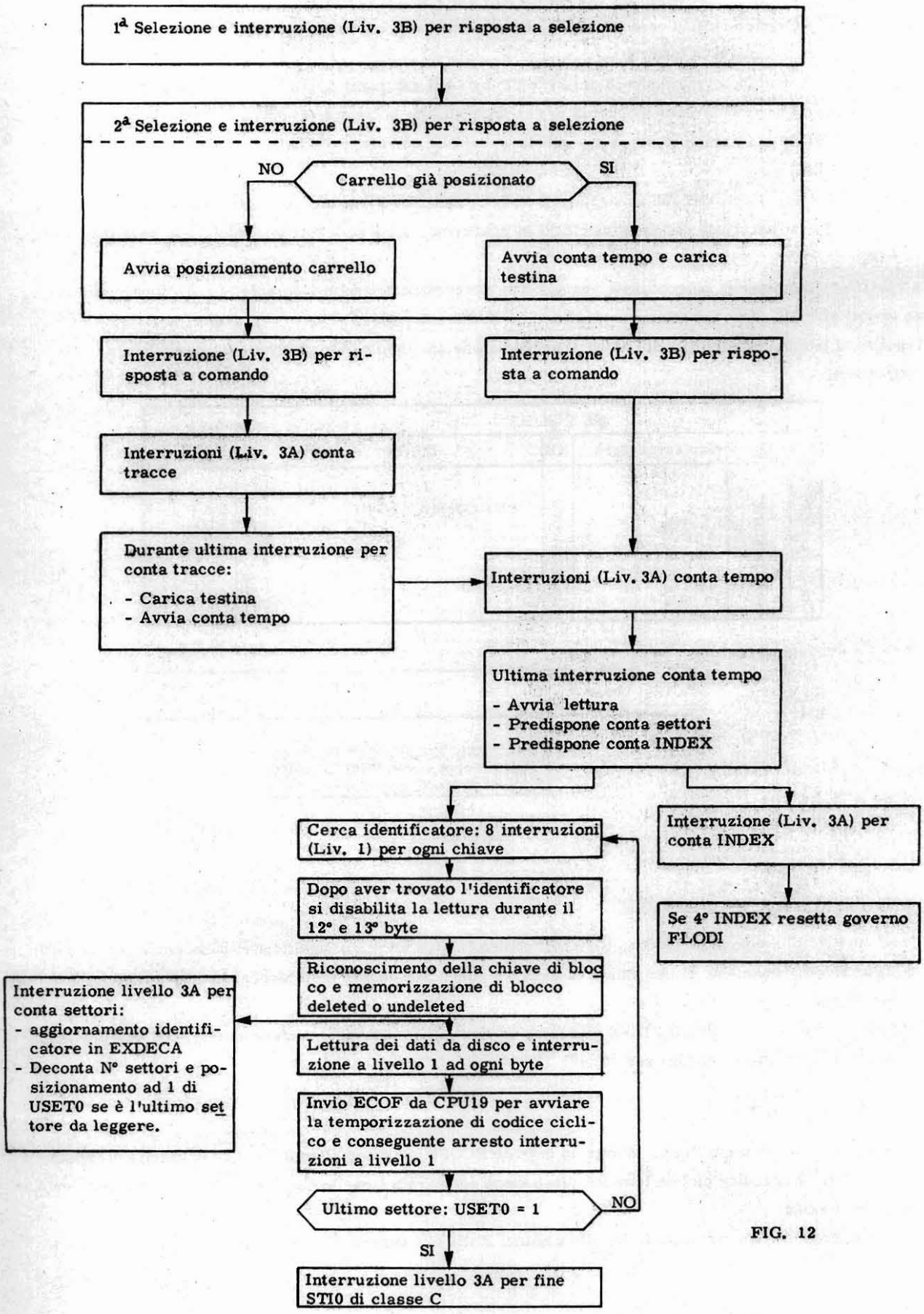


FIG. 12

- FUNZIONAMENTO DELLA SCANSIONE -

L'operazione di scansione è sempre preceduta da un eventuale posizionamento e da una conta tempo.
Durante la decima interruzione per conta tempo ricordiamo che l'U. C. emette due comandi, col primo posiziona SCAN0, SPEC0 e REGO0 sulla logica dei comandi e il secondo porta invece sul "buffer LL maschera" la lunghezza delle maschere da confrontare.
In realtà ciò che viene trasferito sul buffer è il complemento a $[FF]_{16}$ della lunghezza della maschera alla quale l'U. C. somma in precedenza un 1; ad esempio: per una lunghezza di 6 byte viene emessa la configurazione $[FA]_{16}$.

- Controllo della lunghezza e del numero delle maschere confrontate

Dopo aver individuato l'identificatore del primo settore da controllare, in corrispondenza della chiave di blocco, il contenuto del "buffer LL della maschera" è travasato sul "contatore LL maschera".
Ad ogni byte confrontato, il contatore avanza di una posizione e quando raggiunge la posizione FF nasce il segnale che indica la fine della maschera (MAM0) e mentre somma un 1 al "contatore delle maschere confrontate" (NUM0 + 7) abilita di nuovo il travaso del "buffer LL chiave" nel contatore lunghezza chiave.
La somma sul contatore delle maschere confrontate è bloccata quando il segnale SCOK0 (scansione verificata è messa in set).

- Confronto delle maschere coi byte letti

I byte letti sono come al solito ritardati su PAR0 + 7 e di lì sono portati ad un confrontatore. L'altro termine di confronto sono i byte emessi da U. C. durante la serie di interruzioni a livello 1 che il FLODI esegue a partire dal ritrovamento della chiave di blocco.
I byte emessi da U, C. sono staticizzati su INP0 + 7 e quindi serializzati su FSC70 che è il secondo ingresso al confrontatore.
Se da FSC0 + 7 transita un byte $[FF]_{16}$ viene decodificato e genera il segnale BLAN0 che blocca il confronto. Il confronto eseguito è funzione della posizione dei F. F. SPEC0 e REGO0, mentre il risultato è sintetizzato sui F. F. DIVE e SCOK0.
SCOK0 ad 1 indica che la scansione è verificata mentre DIVE0 sta ad indicare che la scansione si è verificata sulla condizione di diverso ($\gtrless$).
In figura 11 è sintetizzato un caso di SCAN (=) con maschera di 6 byte; come si può vedere
DIVE è settato ogni volta che vengono confrontati due byte diversi ed è comunque resettato alla fine del confronto di ogni maschera.
Solo se alla fine di una maschera DIVE0 è a zero viene settato SCOK0 che arresta la logica di scansione e genera la segnalazione di ultimo settore da trattare.
In caso di scansione su $\geqslant$ o su $\leqslant$ si controllano i primi due bit diversi e se c'è la condizione richiesta si setta SCOK0 mentre DIVE0 impedisce il set di SCOK0 coi bit successivi ai primi due diversi.

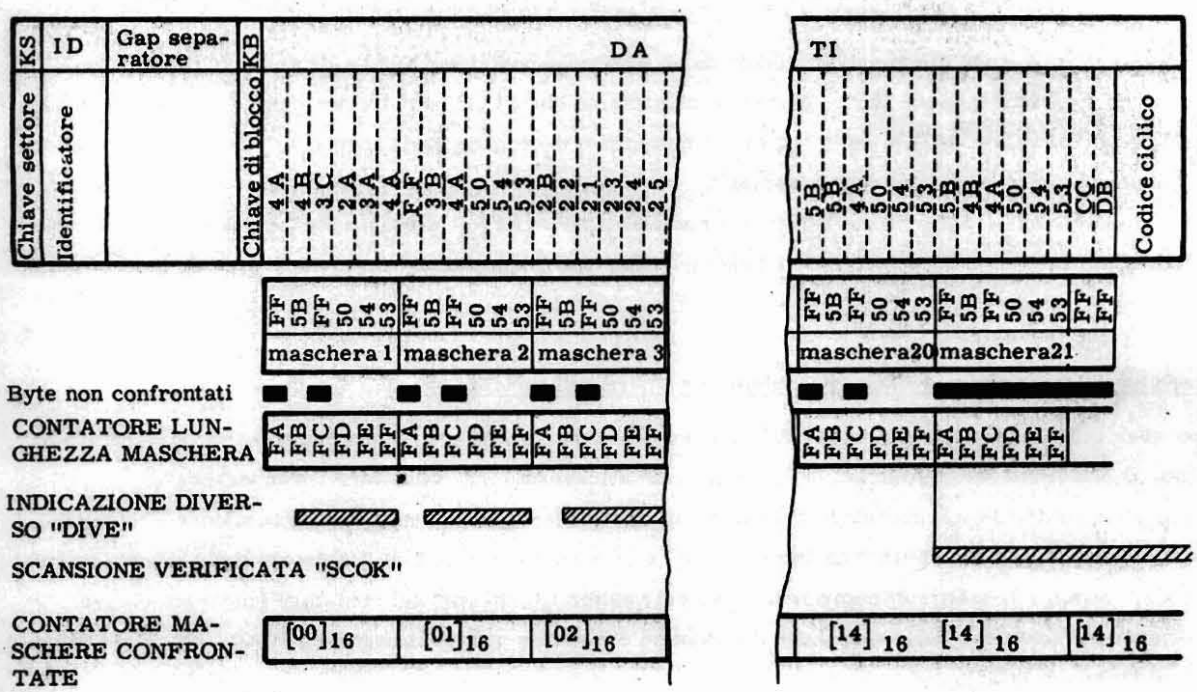


FIG. 13

- Controllo dei settori da trattare in scansione

Quando viene individuata la chiave di blocco di un settore su cui va eseguita la scansione, il FLODI esegue come sempre una interruzione a livello 3A per la conta dei settori trattati e l'U. C. conta i settori trattati ed aggiorna l'indirizzo del settore da trattare successivamente tenendo conto del passo programmato.

Se durante il settore SCOK0 va in set il FLODI predispone la segnalazione di ultimo settore da trattare, se invece durante la interruzione per conta settori trattati l'U. C. rileva che si è sull'ultimo settore essa stessa dà la segnalazione.

Durante l'interruzione di fine, l'U. C. introduce una prima volta il byte di stato e rileva le condizioni del FLODI, esegue quindi due micro istruzioni CAE che settano PRICO e quando va a introdurre per la seconda volta il bus di stato riceve il contenuto del contatore NUM0 ÷ 7

FLUSSO DELLA LOGICA DELLA SCANSIONE

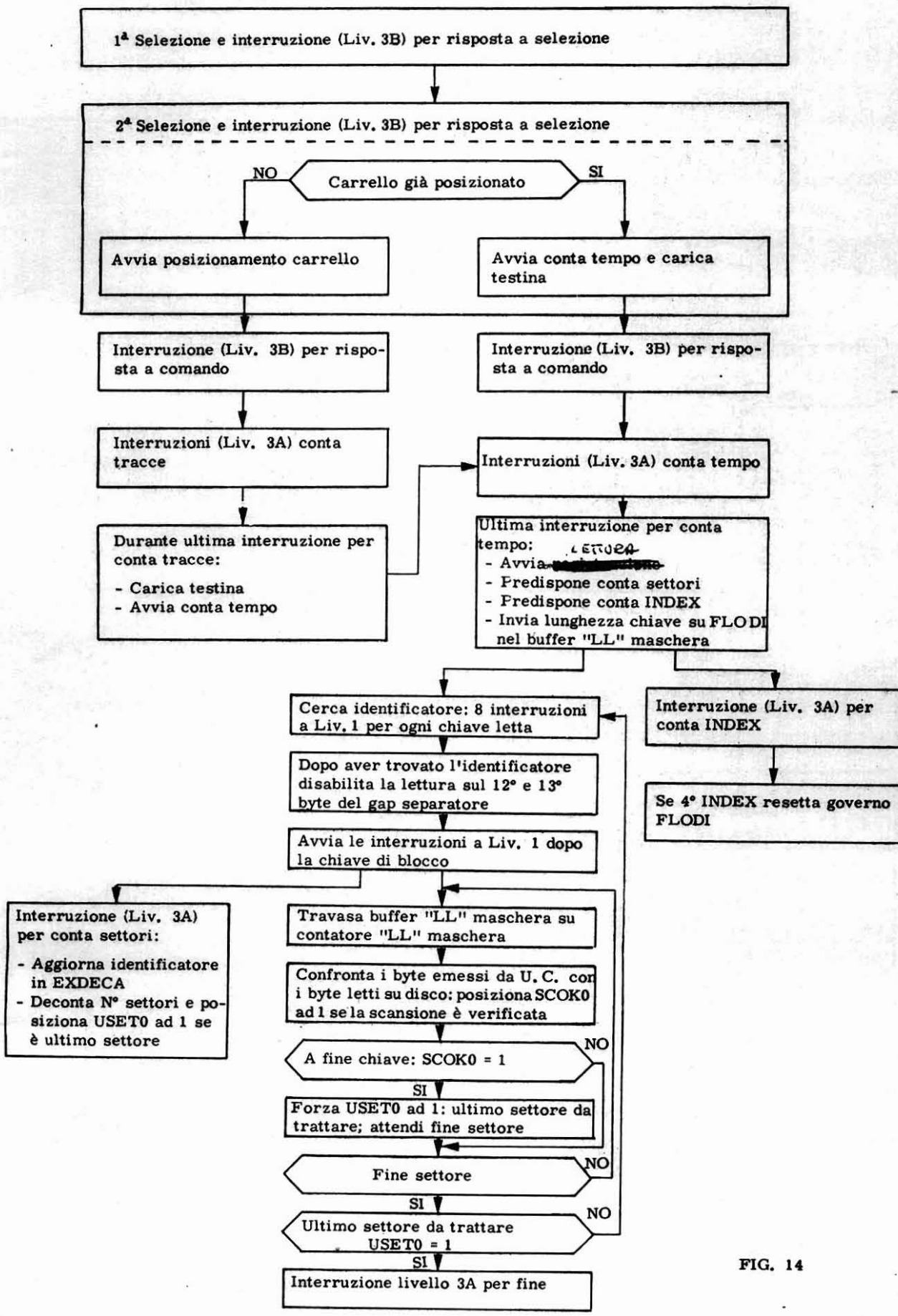


FIG. 14

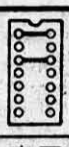
TABELLE RIASSUNTIVE

Inseriamo a questo punto del libro una serie di tabelle riassuntive alle quali ci si riferirà durante lo studio della logica.

- Tabella dei codici di selezione e del bus nome

La presente tabella fornisce per ogni driver i nomi software con cui possono essere selezionati, il byte emesso sul bus dati con la ESE di selezione, i ponticelli da fare perchè il FLODI risponda ad un certo nome, il bus nome emesso e il ponticello da fare perchè il FLODI emetta il corretto bus nome.

TABELLA DEI CODICI DI SELEZIONE E DEL BUS NOME

DRIVER	Nome software	Bus selezione								Bus nome								PONTICELLI
		7N	6N	5N	4N	3N	2N	1N	0N	7N	6N	5N	4N	3N	2N	1N	0N	
1	C.0	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	
2	C.1	0	0	0	1	1	1	1	1									
1	D.0	1	0	0	1	0	1	1	1	1	0	0	1	0	1	1	1	
2	D.1	0	0	0	1	0	1	1	1									

Tab. 2

- Bus tipo

Il bus tipo definisce il tipo di interruzione in atto. Qui ci limitiamo per ogni livello di interruzione a riportare il bus emesso su EPT0N ÷ 7N

Tab. 3

BUS TIPO

Liv. interruz.	Descrizione della interruzione	BUS TIPO SU EPT0N							
		7N	6N	5N	4N	3N	2N	1N	0N
1	Scambio dati	1	1	1	1	1	1	1	*
3A	Fine comando classe C	1	1	1	1	1	0	1	1
3A	Funzione di governo	1	1	1	1	0	1	1	1
3B	Risposta a comando	1	1	1	1	1	1	0	0
3B	Risposta a selezione	1	1	1	1	1	1	1	0

* Viene emesso 0 se ER0R0 = 1, ovvero se c'è errore di codice ciclico o non è ritrovato l'identificatore voluto.

- Bus dati da FLODI (tabella 4)

Sul bus EPDO0 ÷ 70 sono emessi diversi tipi di informazione in funzione del tipo di interruzione.
Solo durante l'interruzione per fine comando di classe "C" vengono emesse due diverse informazioni.
All'inizio della interruzione sul bus dati é emesso lo stato macchina e dopo che l'U. C. ha eseguito una microistruzione per l'emissione di comandi sul bus compare il buffer NUM0 ÷ 7 che riporta il numero di maschere confrontate prima che la scansione desse esito positivo nell'ambito del settore.

- Tabella dei byte di comando emessi da CPU 19

Il byte emesso é funzione del tipo di operazione che si sta eseguendo e del tipo di interruzione in atto.
Nella tabella 5 per ogni caso é indicato il byte emesso. nell'ordine su ECD70, ECD60, ECD50... ECD0 e i F. F. o segnali posizionati da ogni bit.

SIGNIFICATO DEI BIT EPD0 ÷ 7 NEI VARI CASI						
BIT	SCAMBIO CARATTERI	RISPOSTA A SELEZ.	RISPOSTA A COMANDO	FUNZIONE GOVERNO	FINE COMANDO CLASSE "C"	
					1° INPUT (PRIC0=0)	2° INPUT (PRIC0=1)
EPD00	OUT00			IND0 : INDEX	IND0 : INDEX	NUM00
EPD10	OUT10	GOC0 - OR logico di: - Via carrello - Testina abbassata - Cambio disco	GOC0 - OR logico di: - Via carrello - Testina abbassata - Cambio disco	COTE: Conta tempo	COTE: Conta tempo	NUM10
EPD20	OUT20	GOC0 - OR logico di: - Via carrello - testina abbassata - Cambio disco	GOC0 - OR logico di: - Via carrello - Testina abbassata - Cambia disco	TEVE: Scansione verificata su diverso	TEVE: Scansione verificata su diverso	NUM20
EPD30	OUT30	INOP: Intervento operatore	INOP: Intervento operatore	SCOK: Scansione verificata	SCOK: Scansione verificata	NUM30
EPD40	OUT40	LOCA: DRIVER selezionato in locale	LOCA: DRIVER selezionato in locale			NUM40
EPD50	OUT50	MADI: Macchina disponibile	MADI: Macchina disponibile	MADI: Macchina disponibile	MADI: Macchina disponibile	NUM50
EPD60	OUT60	PIZE: Carrello in traccia 00	PIZE: Carrello in traccia 00	PIFU = PIZE: Carrello in traccia 00	PIFU = ERR0: Errore in codice ciclico	NUM60
EPD70	OUT70	DIPE: Disco non presente o sportello aperto	DICA: Disco non inserito o sportello aperto	ERAS: Blocco deleted	ERAS: Blocco deleted	NUM70

Tab. 4

COMANDI EMESSI E SEGNALI PREDISPOSTI SU FLODI

	OPERAZIONE STI0 O SI0	CODICE SOFTWARE	CLASSE COMANDO	RISPOSTA ALLA 2ª SELEZIONE		ULTIMA FUNZIONE DI GOVERNO DI POSIZIONAMENTO CARRELLO	ULTIMA FUNZIONE DI GOVERNO PER CONTA TEMPO O PER CONTA SETTORI		INTERRUZIONE DI FINE COMANDO DI CLASSE "C" O FINE CONTA INDEX
				1° comando PRIC0 = 0	2° comando PRICO = 1		1° Comando PRIC0 = 0	2° Comando PRIC0 = 1	
1	POSIZIONAMENTO SU UNA TRACCIA QUALSIASI	4.4	B	0100 x 010 ECD10-SPEC0 ECD30-DIRI0 ECD40-CLAB0 NOTA 1	0101 x 010 ECD10-SPEC0 ECD30-DIRI0 ECD40-VIRI0 ECD60-CLAB0	11 00 00 00 ECD60-CLAB0 ECD70-CATE0	00000000		
2	POSIZIONAMENTO SU TRACCIA "00"	4.8	B	NOTA 2 come sopra	come sopra	come sopra	come sopra		
3	LOCALE			00 000000	0000 0000				
4	CAMBIA DISCO			0110 x 010 ECD5 - VIDI0 ECD6 - CLAB0	0110 x 010 ECD5 - VIDI0 ECD6 - CLAB0				
5	POSIZIONAMENTO E LETTURA	C. 0	C	0100 x 000 ECD30-DIRI0 ECD60-CLAB0 NOTA 1	0101 x 000 ECD30-DIRI0 ECD40-VIRI0 ECD60-CLAB0	10 00 0000 ECD7 - CATE0	1000 0000 ECD7 - CATE0	0000 0000 MAS0 ÷ 7 [00] 16	0000 0000
6	POSIZIONAMENTO E REGISTRAZIONE BLOCCHI DELETED O UNDELETED	8.1 8.0	C C	come sopra NOTA 1	come sopra	10 00 0001 ECD70 - CATE0 ECD00 - REGO0	10 00 0001 ECD70 - CATE0 ECDO0 - REGO0	0000 0000 MAS0 ÷ 7 [00] 16	0000 0000
7	POSIZIONAMENTO E VERIFICA	8.2	C	come sopra NOTA 1	come sopra	1000 0010 ECD10 - SPEC0 ECD70 - CATE0	1000 0010 ECD10 - SPEC0 ECD70 - CATE0	0000 0000 MAS0 ÷ 7 [00] 16	0 000 0000
8	POSIZIONAMENTO E SCANSIONE SU ⩾	8.6	C	come sopra NOTA 1	come sopra	1000 0110 ECD10 - SPEC0 ECD20 - SCAN0 ECD70 - CATE0	1000 0110 ECD10 - SPEC0 ECD20 - SCAN0 ECD70 - CATE0	xxxx xxxx MAS0 ÷ 7 [xxxx xxxx]	0000 0000
9	POSIZIONAMENTO E SCANSIONE SU =	8.4	C	come sopra NOTA 1	come sopra	10 000 100 ECD20 - SCAN0 ECD70 - CATE0	1000 0100 ECD20 - SCAN0 ECD70 - CATE0	xxxxxxxx MAS0 ÷ 7 [xxxxxxxx]	0000 0000
10	POSIZIONAMENTO E SCANSIONE SU ⩽	8.5	C	come sopra NOTA 1	come sopra	1000 01 01 ECD00 - REGO0 ECD20 - SCAN0 ECD70 - CATE0	1000 0101 ECDO0 - REGO0 ECD20 - SCAN0 ECD70 - CATE0	xxxxxxxx MAS0 ÷ 7 [xxxxxxxx]	0000 0000
11	POSIZIONAMENTO E REGISTRA TRACCIA	8.3	C	come sopra NOTA 1	come sopra	1000 0011 ECD00 - REGO0 ECD10 - SPEC0 ECD70 - CATE0	1000 0011 ECD00- REGO0 ECD10- SPEC0 ECD70- CATE0	0000 0000	0000 0000

NOTA 1 - Nel caso in cui la testina sia già posizionata il comando emesso è quello che normalmente viene emesso sulla ultima funzione di governo per posizionamento carrello.

NOTA 2 - Anche se la testina è già in traccia zero si dà il via al carrello e si ha un PAVAA

Tab. 5

DESCRIZIONE DELLA LOGICA DI FUNZIONAMENTO

Consigliamo di seguire lo studio della logica seguendo oltre che gli schemi logici anche lo schema a blocchi di figura 2.
Descrivendo il funzionamento dei circuiti ci preoccuperemo di fornire anche le indicazioni per il ritrovamento dei segnali sugli schemi logici (DISL).

- LOGICA DELLA SELEZIONE E DELLA RISPOSTA A SELEZIONE -

Avendo esaminato già in precedenza il principio di funzionamento della selezione, di seguito ci limiteremo a spiegare strettamente i circuiti che realizzano la logica.

- Posizionamento di SELE0 e SEDI0

L'U. C. eseguendo la microistruzione ESE per la selezione, a seconda del nome software programmato, emette un certo byte sul bus dati accompagnato da ECOT ed ECOS (tabella 2).
Se il FLODI ha una ponticellatura adeguata,il byte emesso porta il segnale SELSN (DISL 002. A2) a zero e quindi abilita il set di SELE0 (DISL02. A3) che avverrà con lo strobe ECM3.
Contemporaneamente a SELE0, viene posizionato anche il F. F. SEDI0 (DISL 002. A5) che seleziona uno dei due trascinatori sull'FDU.
SEDI0 viene commutato solo se non é già avviata un'operazione di cambio disco o di trattamento del disco ovvero se GOCON é ad 1.

- Interruzione per risposta a selezione

SELE0 provoca una richiesta d'interruzione a livello 3B; SELE0 é già sincronizzato con un ECM3 e genera direttamente RIMU0 (DISLO06. A8) che genera a sua volta EPRBN (DISL007-U5).
Quando l'unità centrale accetta l'interruzione a livello 3B emette ECCB0 che nella logica delle interruzioni genera RIMI0 (ECZON permettendo, ovvero se non c'é richiesta da governi più prioritari).
RIMI0 assume pertanto il significato di interruzione a livello 3B in atto.

- Bus di stato

Il FLODI risponde alla selezione inviando sul bus dati lo stato del sottosistema selezionato (tabella 4).
Si noti che se la selezione é arrivata mentre il driver non selezionato stava lavorando o se era in atto un'operazione di cambio disco, il F. F. SEDI0 non é commutato,perciò sullo stato macchina sarà inviata l'informazione di sottosistema non disponibile col segnale GOCON (DISL02. P7).
La rete del bus dati può ripetere i segnali STA0 + 7 o STB0 + 7, nel nostro caso lavoreranno i segnali STA0 + 7 e precisamente gli ingressi in basso di ogni sezione dei "2 INPUT MULTIPLEXER".

- Bus nome

Il FLODI risponde alla selezione emettendo sul bus nome (DISL07) il proprio nome (tabella 2).
Il nome emesso varia a secondo che sia presente o no un ponticello tra massa e PONO0(DISL 002-A1)

- Bus tipo

Il bus emesso é riportato nella tabella 3 e i circuiti logici interessati sono sul DISL 007.

- 1ª Selezione

Con la prima selezione l'unità centrale si limita ad introdurre i tre bus stato, tipo e nome e decide se proseguire o no l'operazione. La interruzione termina quando l'U. C. emette un ECM3 che resetta SELE0 e fa quindi salire EPRBN.

- 2ª Selezione e comandi relativi

Dopo aver introdotto i tre bus di stato, tipo e nome, se il sottosistema é disponibile l'U. C. emette il comando di cambia disco, di poni in locale o una coppia di comandi per avviare il carrello.
Ogni comando viene emesso sul bus ECD0 + 7 accompagnato dagli strobe ECOC ed ECOT.
Il primo comando emesso posiziona il F. F. PRIC0 (DISL 002) che sarà già in set all'arrivo del secondo comando, anche se non verrà utilizzato.
I comandi emessi sono riportati tutti nella tabella 5.

- Comando di locale

Dei comandi emessi durante la selezione l'unico che abbia ECD60 ed ECD70 a zero é il comando di poni in locale che setterà i F. F. LOC1 o LOC2 (DISL002. M3 e P3) a seconda che sia selezionato il driver 1 o 2 ovvero a seconda che SEDIA sia a zero o ad 1.
Lo stato di locale cessa o con un reset da governo ECOR B o con la caduta di MADI1 ovvero aprendo lo sportello dell'FDU.

- Cambia disco

Il comando di cambia disco viene eseguito solo se é stato selezionato il driver 2. In questo caso il bit ECD5 può generare il segnale SED50 (DISL 02, G7) che lavora in ingresso ai due "4 bit shift register" usati per la staticizzazione dei comandi.
Emettendo il comando nasce il segnale MACON (DISL02-C8) abilitato da RIDEN perché RIMI0 (interruzione a livello 3 B) e COLON sono ad 1.
MACON carica, sugli shift register, ECD5 e nasce quindi il comando VIDI0 che é rigenerato (DISL 09) col nome CADIA che é il comando di cambia disco.

- Comando del senso di avanzamento del carrello (1° comando)

Il comando MACON, come si é già visto nella logica del comando cambia disco, carica la configurazione ECD0 + 7 sui "4 shift register" di staticizzazione. In questo modo viene posizionato il segnale DIRI0 che su FDU determina il verso di avanzamento.

- Comando di avanzamento carrello (2° comando)

Il comando viene anch'esso staticizzato sugli staticizzatori e mentre si conferma DIRI0 viene posizionato anche il comando VIRI0 di via carrello.

- LOGICA DELL'INTERRUZIONE PER RISPOSTA A COMANDO -

Gli ECOC emessi durante una interruzione per risposta a selezione (RIMI0 ad 1) predispongono il set temporizzato del F. F. COMA0 (DISL02. E3) che verrà definitivamente settato nel momento in cui viene emesso l'ECM3 che resetta la interruzione per risposta a selezione.
COMA0 come già SELE0 genera RIMU0 e quindi EPRBN per una richiesta interruzione a livello 3 B.
Quando l'interruzione é accettata viene inviato all'U. C. il bus tipo, nome e stato.
Il bus tipo (tabella 3) dice che si tratta di risposta a comando. Il bus nome già visto per la risposta a selezione dice qual'é l'unità che ha interrotto mentre il bus di stato porta le stesse informazioni già inviate con la selezione.
L'interruzione termina quando l'U. C. emette un ECM3 che resetta il F. F. COMA0.

- LOGICA DI POSIZIONAMENTO DEL CARRELLO -

Si é già visto come durante la seconda selezione col primo comando si predispone il verso d'avanzamento e col secondo comando si conferma il verso e si avvia il carrello.
Alla fine della seconda selezione nasce la richiesta di interruzione a livello 3 B per risposta a comando.

- Richiesta di interruzione per conta tracce

Una volta predisposto VIRI0 e DIRI0 l'avanzamento del carrello ha inizio e procede con tempi proprii.
Ad ogni traccia, l'FDU emette un impulso sul filo PAVAA che nella "logica delle interruzioni" pone in set il F. F. FOGO0 (DISL006. A4) che é la stazione asincrona delle richieste d'interruzione per funzioni di governo.
FOGO sincronizzato da ECM3 setta FUGO0 (DISL006. C6) che genera EPRA ed emette una richiesta di interruzione a livello 3 A. Quando l'U. C. emette il segnale ECCA0 per dire che la interruzione é stata accettata, nasce il segnale MEMA0 che ha il senso di interruzione a livello 3 A in atto.
Durante l'interruzione il FLODI emette il suo nome sul bus nome, sul bus tipo l'informazione che si tratta di interruzione per funzione di governo e predispone lo stato del FLODI sul bus EPD0 ÷ 7.
Questa volta su EPD0 ÷ 7 saranno abilitati STB0 ÷ 7 (DISL007 ed esattamente gli ingressi in alto dei circuiti "2 INPUT MULTIPLEXER" (DISL 007).
L'U. C. esaminando il bus di stato (tabella 4) vede che non si tratta di funzione per conta tempo (COTE0 = 0) o di conta INDEX (INDO0 = 0) e per esclusione stabilisce che si tratta di conta tracce per il posizionamento.
L'U. C. normalmente deconta i passi avvenuti e può quindi sapere quando il carrello ha raggiunto la posizione voluta. Nel caso di comando di ritorno carrello in traccia "00" (HOME) l'U. C. controllando il bus di stato ed esattamente controllando PIZE (carrello in traccia "00") sa se il posizionamento é finito.
Finché il carrello non ha raggiunto la posizione voluta, l'U. C. mentre introduce il bus tipo, nome e stato emette un ECOT che resetta FOGO0 e successivamente emette un ECM3 che resetta anche FUGO0 e tronca quindi la interruzione per funzione di governo.

- Fine posizionamento

Quando durante una interruzione l'U. C. si rende conto che é stata raggiunta la traccia voluta, emette un comando sul bus ECD0 ÷ 7 accompagnato da ECOC ed ECOT. Il comando emesso é un comando di avviamento della conta tempo che é sempre necessaria dopo un movimento del carrello il quale impiega un certo tempo a stabilizzarsi. Il comando di conta tempo può anche comprendere il comando di "carica testina" se al posizionamento segue un trattamento dei dati su disco (vedi tabella 5).

La conta tempo è avviata generando il segnale CATE0 (DISL002. E9) sugli staticizzatori dei comandi. CATE0 è accompagnato da CLABN se deve essere caricata la testina. CATE0 passando fisicamente da una piastra all'altra del FLODI acquista il nome di CATE5 e con CLABN genera CACL0 (DISL002. M7) che su FDU determina il caricamento della testina.
CATE0 setta sempre comunque il F. F. COTE0 (DISL002. P9) con senso di conta tempo in atto.
Dopo aver emesso il comando l'U. C. emette sempre l'ECM3 che tronca l'interruzione. Il comando emesso durante una interruzione per funzione di governo non genera una interruzione per risposta a comando come nel caso di comandi emessi durante la risposta a selezione, perchè RIMI0 è a zero e non può settare il F. F. COMA0.

- LOGICA DI CONTA TEMPO -

L'avviamento della conta tempo ovvero il posizionamento di CATE0 e COTE0 è già stato esaminato sulla logica di fine di posizionamento.

- Temporizzazione

Se non si opera su apparecchiature diagnostiche i segnali ECODN ed ECRDN sono ad 1 ed il segnale MH160 ripete il segnale prodotto dall'oscillatore del FLODI che opera alla frequenza di 16 MHz (DISL01. C2).
Essendo CATE0 e COTE0 ad 1 i due "UP binary counter" avanzano e generano gli impulsi TC01 e TC02 con frequenza rispettivamente di 1 MHz e 62,5 KHz (DISL 001. 63 e 65).
Il TC02 a sua volta fa avanzare altri due "UP binary counter" da cui nasce il segnale MCOT0 con periodo di 4 msec e durata 16 μsec.

- Interruzioni per conta tempo

In presenza di COTE0, ogni MCOT0 setta FOGO0 (DISL 06. A4) che come abbiamo già visto nel caso di posizionamento avvia la logica di interruzione per funzione di governo.
L'U. C. ad ogni interruzione deconta un contatore interno che era stato posizionato all'inizio in posizione 10.
Introducendo i byte di tipo, stato e nome l'U. C. emette un ECOT che resetta FOGO0 ed arresta l'interruzione emettendo un ECM3.

- Fine conta tempo

Durante la decima interruzione di conta tempo l'U. C. emette due comandi sul bus ECD0 + 7 accompagnati da ECOT ed ECOC.
Il primo comando come già i comandi emessi in selezione viene staticizzato sui due "4 bit shift register".
Quando al posizionamento deve seguire un trattamento dei dati del disco il comando emesso conferma CATE0 e CLABN ad 1 e posiziona, REGO0, SCAN0 e SPEC0 a seconda del tipo di trattamento da fare.
Il primo comando emesso setta tra l'altro il F. F. PRIC0 (DISL 02. A7) così che quando arriva il secondo comando non può essere caricato sullo staticizzatore per mancanza del clock MACON inibito per l'appunto da PRIC0.
Il secondo comando troverà invece abilitato il clock sul buffer MAS0 + 7 e verrà qui staticizzato.
Il secondo comando emesso, si ricorda, che ha senso solo nelle operazioni di scansione ed è il complemento ad $[FF]_{16}$ della lunghezza della maschera di scansione.

- LOGICA DI CONTA DEGLI INDEX -

Il segnale INDEA generato ad ogni giro del disco selezionato, genera sempre sul FLODI il segnale INDO0 (DISL004. G1) con l'unica eccezione del tempo di stabilizzazione del carrello dati in cui è inibito da COTEN = 0.

INDO0 setta il F. F. FOGO0 (DISL006. A4) che è la stazione asincrona di richiesta interruzione per funzione di governo. FOGO0 setta FUGO0 che é la stazione sincrona e che genera EPRAN (DISL 007 . S5).

Quando l'U. C. accetta la richiesta interruzione nasce MEMA0 (DISL 006. G8) e viene abilitata l'emissione dei bus di stato, tipo e nome come si é già visto nelle interruzioni per conta tempo.

L'emissione di un ECOT durante MEMA0 resetta INDO0 e FOGO0, un ECM3 resetta invece FUGO0 ponendo così termine all'interruzione. Durante le interruzioni per conta index che seguono la conta tempo, l'U. C. può contare i giri del disco ed interrompere le operazioni di trattamento del disco che non terminino entro il terzo giro completo.

- LOGICA DELLA RICERCA DELL'IDENTIFICATORE -

- Abilitazione della lettura

Con tutte le operazioni di trattamento dei dati del disco, all'infuori dell'operazione di registra traccia, nasce il segnale ORLE0 (DISL 002. T9) che predispone l'FDU alla lettura.

In lettura l'FDU, dopo aver sincronizzato il suo discriminatore, invia due mastri: BILEA e COLEA.

Il mastro BILEA ripete i bit d'informazione e COLEA i clock.

- Temporizzazione della lettura

La temporizzazione del FLODI, durante la lettura, non segue il mastro derivato dall'oscillatore, bensì la temporizzazione data dall'FDU col segnale COLE. COLE genera infatti CBIT(DISL03. E4) che è appunto il mastro di temporizzazione a livello bit. Si trascuri al momento il segnale BINCN.

CBTN è il clock sulla temporizzazione a livello byte (CBI0 ÷ 4, DSL003. A7) dalla quale si deriva DEC7 (DSL03/A9) esce alla fine di ogni byte letto. La temporizzazione a livello byte per poter essere sincronizzata coi byte letti è bloccata in configurazione 5 da ABCO0 finchè non viene individuato un byte con clock mancanti che porterà ABTEE a zero. (ABTEE è ABTEN che cambia nome cambiando piastra).

- Riconoscimento di un carattere con missing clock

Ci si riferisca alla temporizzazione di fig. 15. Il segnale INCL (DISL03. A2) si setta con tutti i clock e si resetta con tutti i bit. Rimane quindi a zero per un periodo superiore a 4 µsec (tempo che intercorre tra un clock e il successivo) solo nel caso che ci siano dei clock mancanti.

Il segnale INCL controlla il passaggio di BILE su CBITN, per cui si può dire che su CBITN sono abilitati tutti i COLE ed i BILE preceduti da clock mancanti: ciò permette di avere comunque 8 CBITN per ogni byte e in particolare anche nei byte con missing clock.

Lo shift register FSC0 ÷ 7 (DISL 004. C8) avanza ad ogni CBITN e viene caricato di tutti 1 finché ci sono tutti i clock, viene caricato con zeri solo nel caso di clock mancanti (figura 15).

Una particolare condizione di FSC0 ÷ FSC5, ottenibile solo se sono stati letti dei byte con tre clock mancanti successivi, abilita un impulso sul segnale MISSN (DISL 003. M3).

Il segnale MISSN setta il F. F. ADMA0 (DISL 003. M4) che setta a sua volta il F. F. ABTE0 (DISL 03. E7).

Il segnale ABTEN cambia nome sul pannello posteriore, e diventa ABTEE che portando ABCO0 ad 1 libera il contatore della temporizzazione a livello byte.

Nascono così i segnali DEC7 che coincidono col primo clock di ogni byte.

Il segnale ADMA0 viene resettato dal 2° DEC7.

TEMPORIZZAZIONE
Riconoscimento Byte con Missing Clock

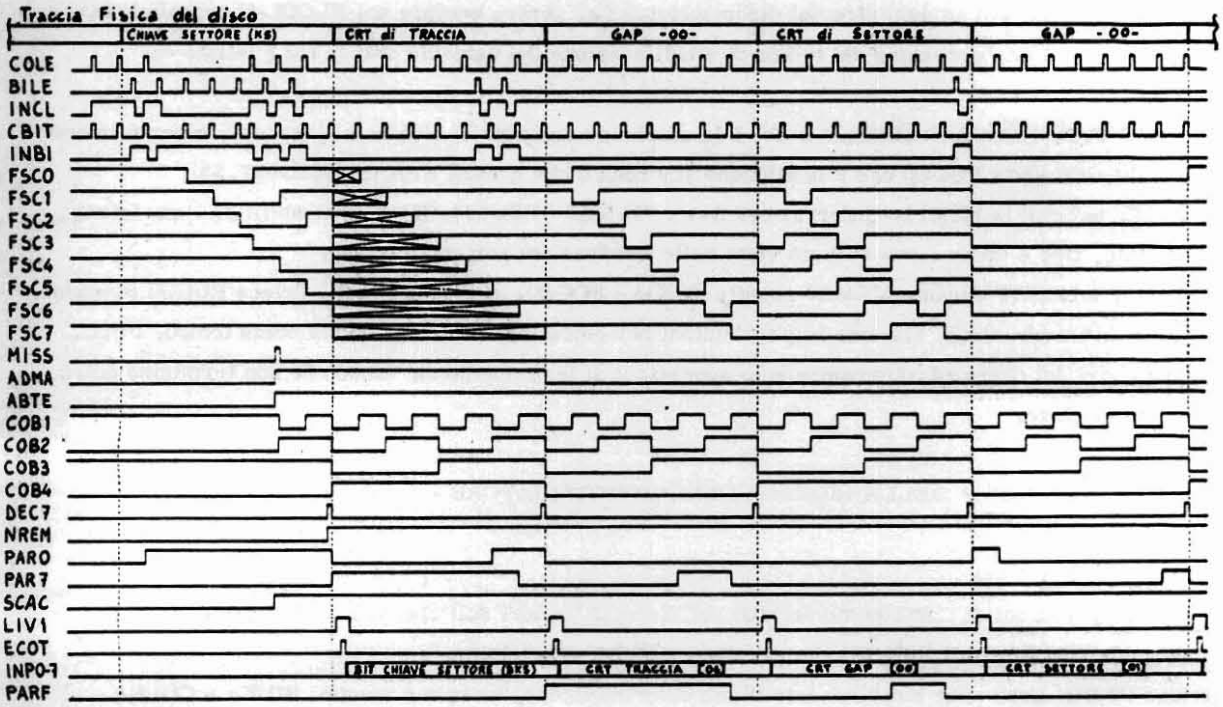


FIG. 15

- Memorizzazione dei dati letti

Riferirsi alle forme d'onda di figura 15.

I bit letti vengono dapprima staticizzati su INBI0 e di qui sono fatti shiftare prima su PAR0 + 7 (DISL004. G6) e quindi su PARF0 (DISL 004. P8).

Nella figura 15 sono riportati in dettaglio le forme d'onda del passaggio dei bit della chiave di settore (BKS) sui buffer sopra menzionati. Nella figura 16 sono invece indicati con maggior dettaglio i contenuti dei buffer con riferimento all'intero identificatore.

- Richiesta all'U. C. dei byte da confrontare coi byte letti

Riferirsi alle forme d'onda di figura 15 e 16.

Il segnale ADMA0 setta SCAC0 (DISL 006. T8) che sta ad 1 per tutto il tempo durante il quale si vuole fare uno scambio dati con l'unità centrale.

SCAC0 abilita ad ogni DEC7, quindi ad ogni byte, il set di RILI0 (DISL 006. M6) che é la stazione asincrona delle richieste di interruzione a livello 1. RILI0 sincronizzato da ECMIN fa settare LIVI0 (DISL 006. P6) che é la stazione sincrona delle interruzioni a livello 1 e genera EPRIN (DISL 007. R5); LIVI0 fa resettare nel frattempo RILI0.

Quando l'interruzione è accettata, l'U. C. emette ECCI0 e fa nascere il segnale RICA0 (DISL 006. M9) sul FLODI. Il FLODI emette quindi i bus nome e tipo (tabelle 2 e 3). L'U. C. ad ogni interruzione emette sul bus dati un byte accompagnato da ECOT.

L'U. C. tronca l'interruzione emettendo un ECM1N che provoca il reset di LIVI0.

Finché SCAC0 resta ad 1, s'innesca una nuova richiesta d'interruzione ad ogni DEC7.

- Memorizzazione dei dati emessi da U. C.

Ad ogni interruzione a livello 1 l'U. C. emette un dato accompagnato da ECOT. Il dato viene caricato in parallelo e staticizzato sul buffer INP0 + 7 e al DEC7 successivo il contenuto di INP0 + 7 é travasato in parallelo su FSC0 + 7 che shifta ad ogni bit.

Mentre FSC0 + 7 shifta, in coda al dato proveniente da U. C., entra su di lui anche il dato letto tramite 1NCL0, ma questo dato viene perso nel momento in cui su FSC0 + 7 é travasato il contenuto di INP0 + 7.

L'U. C. emette nell'ordine i seguenti dati:

1ª interruzione - BKS (configurazione dei bit della chiave di settore)
2ª " - Traccia su cui cercare il settore
3ª " - $[00]_{16}$
4ª " - Settore su cui si vuol operare
5ª " - $[00]_{16}$ contemporaneamente ECOF0 va ad 1
6ª " - CKB (configurazione clock della chiave di blocco)
7ª " - $[00]_{16}$ ECOF0 va a zero
8ª " - $[BKB]_{16}$ (configurazione dei bit della chiave di blocco)

- Temporizzazione del codice ciclico e fine interruzioni a livello 1

Riferirsi alla figura 16.

Il segnale ECOF0, alla fine della interruzione durante il quale viene emesso, provoca il set di FIUC0 (DISL 003. C9) che sta alto fino alla fine della interruzione durante la quale ECOF0 va a 0.

FIUC0 controlla COCI0 (DISL 003. E9) che lo ripete sfasato di un byte. COC10 resetta SCAC0 che provoca l'arresto delle richieste d'interruzione

Le interruzioni che nascono durante la ricerca identificatore sono 8 per ogni carattere con missing clock che viene letto.

- Confronto dei byte letti con quelli emessi da U. C.

Riferirsi alla figura 16.

I primi 5 byte emessi da U. C. vengono confrontati coi bit del carattere con clock mancanti letto e dei 4 byte che la seguono. Il tempo durante il quale si esegue il confronto é quello in cui COID0 (DISL 003. M7) sta alto.

Il confronto é fatto durante COID0 tra i bit uscenti da PARF0 ed FSC7 sulla rete che culmina in ERA0N (DISL 003. P6). Una diversità su un bit qualsiasi porta ERAON a zero e setta EROR0 (DISL 003. P7). EROR0 é messo a zero all'inizio di ogni confronto da EDMAN e rimarrà a zero solo se il confronto é positivo.

- Rete di calcolo e controllo del codice ciclico

La logica descritta di seguito é tutta sul DISL01. Alla fine di ADMA0 il segnale MRESN resetta lo shift register del calcolo del codice ciclico. Lo shift register é composto da 2 "8 bit shift register" e da 1 "4 bit shift register". Dei 20 bit disponibili ne vengono però utilizzati solo 16.

L'ingresso dello shift register è il segnale INCR0, che in lettura è l'OR esclusivo di PAR7 (dato letto) e CRCF che è l'uscita dello stesso shift register.

Il passaggio tra CRC4 e CRC5 è interrotto dalla rete di or esclusivo tra CRC4 e INCR0 come anche il passaggio da CRCB a CRCC é interrotto dalla rete di or esclusivo tra CRCB e INCR0.

Il codice ciclico calcolato tiene conto dei byte letti: "BKS", TRACCIA, $[00]_{16}$, SETTORE e $[00]_{16}$.

Durante COCI0 starà sempre a zero in caso di lettura esatta. Un eventuale bit 1 su INCR0 durante COCI0 posiziona in set EROR0 come se il confronto dei bit dell'identificatore avesse dato esito negativo.

- Segnalazione settore trovato

Durante l'ottava interruzione l'U. C. introduce il bus tipo ed esaminando ETPON (tabella 3) sa se EROR0 é ad 1 o a zero quindi sa se l'identificatore é stato trovato o meno e quindi si predispone per lo scambio dati o per un altro confronto identificatore.

Alla caduta di COCI0 se il settore é stato trovato viene anche settato il F. F. SEOK0 (DISL 003. T2).

- Conta caratteri del gap separatore

SEOK0 a zero forza in reset il contatore MODO0 + 7 (DISL 005. M3 e M4) il quale conterà tutti i DEC7, cioé tutti i byte successivi.

- Memorizzazione dei byte BKB e CKB

Il byte CKB uscendo da FSC7 entra sullo shift register SER7 e se viene settato SEOK0 esso é abilitato e riciclare in continuo su SER7.

Il byte BKB emesso con l'ultima interruzione sul buffer INP0 + 7 rimane su di esso perché mancando altre interruzioni e quindi gli ECOT il suo contenuto non viene alterato e pertanto il byte shifterà sullo shift register FSC0 + 7 in continuo.

In figura 16 sono riportate le forme d'onda relative.

TEMPORIZZAZIONE
Riconoscimento Identificatore

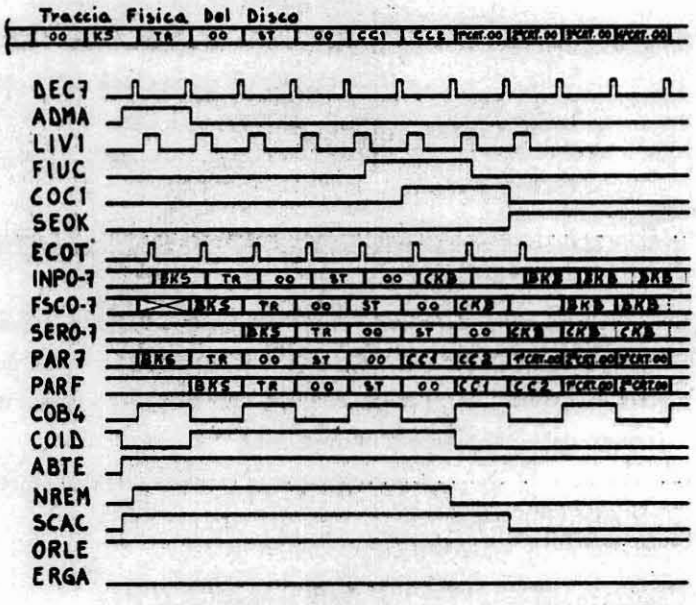


FIG. 16

- LOGICA DELLA LETTURA -

- Logica di aggancio della lettura

La lettura vera e propria con scambio di byte ha inizio col ritrovamento della chiave di blocco, caratterizzata dal "missing clock".

Si ricorda però che sul 10° e 11° byte del gap separatore potrebbero essere registrati dei bit spuri che potrebbero far nascere la segnazione di caratteri con "missing clock".

Al fine di evitare un'errata partenza dello scambio caratteri, si sopprime la lettura per l'11° e 12° byte.

Il contatore MOD0 ÷ 7 controlla il set ed il reset del F. F. ORR0 (DISL 004. E4) che genera ORTE0 (DISL 004. E6) - figura 17 -

ORTE0 blocca ORLE0 (DISL 002. T9) impedendo la lettura e fa sì che la temporizzazione a livello byte venga pilotata dal mastro MOO4 derivato dall'oscillatore di macchina. MOO4 che ha un periodo di 4 μsec si sostituisce così ai clock letti nella generazione del mastro CBIT0 (DISL 003. E4).

E' indispensabile non arrestare la temporizzazione perché MOD0 ÷ 7 possa continuare ad avanzare ad ogni byte del gap separatore. Con la caduta di ORTE0 si blocca la temporizzazione a livello byte.

- Riconoscimento della chiave di blocco e controlli relativi

La logica di riconoscimento di un byte con clock mancanti genera come già si è visto nella ricerca dell'identificatore il segnale MISS verso la fine della chiave. MISS come si è già visto setta ADMA0 che sgancia la temporizzazione di byte e sta in set fino al primo DEC70 successivo al MISS.

Alla fine della lettura del byte con missing clock, ed esattamente quando l'ultimo bit letto é staticizzato su INBI0, il penultimo su PAR0 ed il terz'ultimo su PAR1 nasce il segnale ADCO0 (DISL 003. R8) che abilita l'esame della chiave.

Se il terz'ultimo bit è uno, viene settato il F. F. ERR0 (DISL 003. P8) che verrà inviato sul bus stato alla fine della lettura per indicare una lettura anomala, infatti se il terz'ultimo bit è un uno la chiave letta non può essere una chiave di blocco.

Se il penultimo bit è zero viene settato il F. F. ERAS0 (DISL 003. T8) che verrà inviato sul bus stato alla fine della lettura per indicare che almeno uno dei settori letti era un blocco "deleted".

- Scambio dei byte con l'U. C.

ADMA0 setta NREM0 (DISL 004. A2) che abilita il set di SCAC0 che, come si è già visto, causa ad ogni DEC7 le interruzioni a livello 1. Ad ogni interruzione l'unità centrale introduce il bus dati sul quale è abilitato OUT0 ÷ 7 (DISL 004. G8) che contiene i byte letti.

Per la fasatura tra le richieste interruzioni ed il contenuto di OUT0 ÷ 7 si faccia riferimento alla temporizzazione di figura 17.

Lo scambio caratteri termina colla fine di SCAC0 che cade con la temporizzazione di codice ciclico COCI che segue come al solito ECOF0.

- Temporizzazione di controllo del codice ciclico

L'U. C. con la $128^a/256^a$ interruzione a livello 1 emette ECOF0 che richiamerà alla $130^a/258^a$. ECOF0 controlla COCI0 (DISL 003. E9) che alla caduta resetta SCAC0.

- Calcolo del codice ciclico e suo controllo

Il segnale ADCO0, dopo che è stata letta tutta la chiave di blocco, resetta il buffer di codice ciclico CRC0 ÷ F (DISL 001. Il calcolo del codice ciclico è fatto tenendo conto dei byte in uscita da PAR7, che saranno trattati come è già stato detto nella logica di ricerca identificatore. Un eventuale bit 1 in uscita dalla rete di OR esclusivo INCR0 durante COCI0 setta ERR0 che alla fine della lettura di tutti i settori verrà inviato all'U. C. sul bus di stato.

- Conteggio dei settori letti e segnalazione di ultimo settore

Tutte le volte che si trova una chiave di blocco, nasce il segnale INFSN (DISL 003. U4) che setta il F. F. FOGO0 (DISL 06. A4) che è la stazione sincrona delle interruzioni per funzione di governo. FOGO0 setta FUGO0 (DISL 006. C6) che genera EPRAN (DISL 007. T5). Quando l'interruzione è accettata vengono emessi i bus nome, tipo e stato (tabelle 2, 3 e 4).

L'U. C. prende atto dell'interruzione, conta i settori letti, aggiorna l'identificatore del prossimo settore da leggere e tronca l'interruzione emettendo ECM3N.

Se il settore che si sta per leggere è l'ultimo, l'U. C. emette due comandi sul bus dati accompagnati da ECOC ed ECOT; i due comandi sono identici ai due comandi emessi alla fine della conta tempo e che hanno avviato la lettura.

Il secondo comando emesso genera il segnale 1NCO0 (DISL 005. A2) che genera USET0 (DISL 006 . P1) che è la segnalazione di ultimo settore.

- Logica di fine della lettura di tutti i settori

La caduta di SEOK0, con USET0 alto setta il F. F. FINE0 (DISL 006. E4) che é la stazione asincrona di richiesta interruzione. FINE0 viene sincronizzato da ECM3 e genera RIFI0 che abilita EPRAN (DISL 007. T5). Quando l'interruzione a livello 3A é accettata nasce il segnale MEMA0 (DISL 006. G9). Il FLODI emette quindi il nome, ed il tipo della interruzione secondo le tabelle 2 e 3 Il FLODI emette pure il bus stato su EPD0 + 7 (DISL 007) abilitando in uscita gli ingressi alti dei multiplexer STB0 + 7 col significato indicato nella tabella 4.

L'U. C. dopo aver introdotto il bus stato emette un byte $[00]_{16}$ di comando su ECD0 + 7. Il comando è accompagnato dagli strobe ECOT ed ECOC. Il comando provoca il reset del buffer di staticizzazione dei comandi e quindi viene sollevato il pressore della testina.

TEMPORIZZAZIONE
Lettura Ultimo Settore

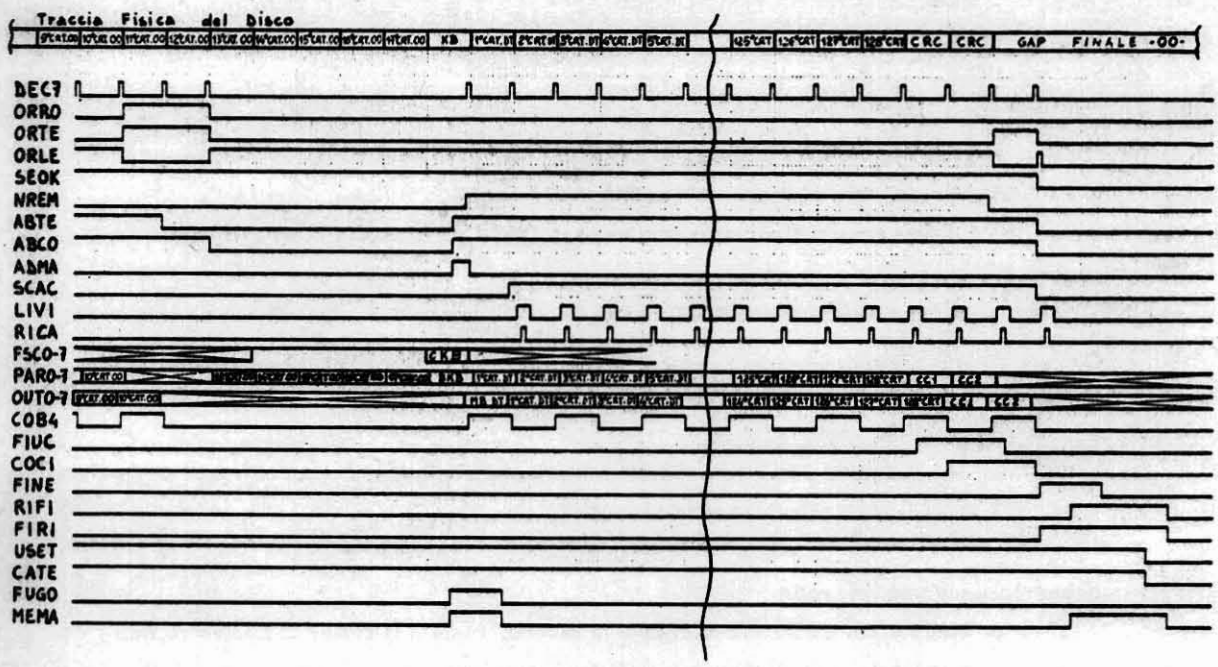


FIG. 17

Il comando emesso da U. C. setta su FLODI il F. F. PRIC0 (DISL 02. A7) che abilita sul bus dati EPD0 ÷ 7, quindi, non più lo stato della macchina, ma il contenuto del buffer NUM0 ÷ 7 (DISL 001. G7 e G9) che non ha senso in lettura. NUM0 ÷ 7 ha senso solo alla fine di una operazione di scansione e dice su quale maschera la scansione si é verificata.
Dopo aver introdotto il secondo byte da EPD0 ÷ 7, l'U. C. emette un ECM3 che arresta l'interruzione per fine.

- LOGICA DELLA REGISTRAZIONE -

Se si avvia un'operazione di registrazione, alla fine della conta tempo, durante la decima interruzione per funzione di governo, l'U. C. emette un comando che viene staticizzato sul buffer dei comandi e pone REG00 ad 1 (DISL 002. E8).
La registrazione prosegue con una ricerca identificatore alla stessa maniera della lettura.

- Logica d'aggancio della registrazione

Riferirsi a figura 18.
La registrazione ha inizio sul decimo byte del carattere separatore; infatti il contatore MOD0 ÷ 7 (DISL 005. M3 e M4) controlla ORR0 (DISL 004. E4) che sta in set per il 10° e 11° carattere e controlla il set di ORRE0 all'inizio dell'11° carattere. ORR0 ed ORRE0 fanno nascere ORTE0 (DISL 004. E5), che blocca ORLE0 (DISL 002. T9) arrestando così la lettura e abilita su CBIT0 (DISL 003. E3) la temporizzazione ricavata dall'oscillatore del FLODI. ORRE0 starà in set per tutta la durata di SEOK0 più ulteriori 8 µsec controllati dalla temporizzazione.

- Controllo della corrente di cancellazione

La corrente di cancellazione dei gap tunnel è controllata da ERGA0 (DISL 004. T4) emesso da FLODI. ERGA è controllato da ORRE0 e dalla temporizzazione del FLODI, il set di ERGA avviene 128 µsec dopo "ORRE0" quando NUMO0 e NUM30 sono a 1, il reset di ERGA avviene invece 464 µsec dopo la caduta di ORRE0.

- Registrazione dei clock

ORRE0 in set abilita la nascita del segnale di registrazione INRE0 (DISL 008. E9). In particolare il segnale INR1N abilita un impulso su INRE0 ad ogni MOO4 e controlla la registrazione dei clock.

- Registrazione della chiave di blocco

Quando il contatore MOD0 ÷ 7 ha raggiunto la posizione 15, ovvero alla fine del gap separatore, nasce il segnale CROF0 (DISL 005. M3) che abilita il set di REAM0 (DISL 004. C4) il quale rimarrà in set per un intero byte. REAM0 abilita il set di NREM0 (DISL 004. A2) che impedisce ulteriori partenze di REAM0 per tutto il settore.
Si ricorda che i byte relativi alla chiave di blocco, sono rimasti imprigionati negli shift register SER7 ed FSC71: esattamente in SER7 ricicla la configurazione dei clock della chiave di blocco (CKB) e in FSC71 passa continuamente la configurazione dei bit della chiave di blocco "BKB" che può essere indifferentemente relativa ad un blocco deleted" o no.
REAM0 abilita sul segnale di registrazione dei clock INR1N la soppressione di alcuni clock secondo la configurazione di SER70. REAM0 setta inoltre AREA0 (DISL 004. P1) che abilita l'uscita di FSC71 sul mastro di registrazione INR3N e pertanto durante REAM0 viene scritto il byte BKB emesso da U. C.

- Logica di scambio dei byte da U. C. verso FLODI

Nel momento stesso in cui si setta REAM0 viene settato il F. F. SCAC0 che come è ormai noto causa le interruzioni a livello 1. Ad ogni interruzione l'U. C. emette un byte che viene staticizzato dapprima su INP0 ÷ 7 poi serializzato su FSC71 da cui passa sul mastro di registrazione (INR3N).

- Fine scambio byte e temporizzazione del codice ciclico

Riferirsi a figura 19.

Durante la 129ª/257ª interruzione a livello 1 l'U. C. emette ECOF0 che sarà richiamato a zero durante la 131ª/259ª interruzione. ECOF0 controlla, come già visto, COCI0 che sta in set per due byte.

COCI0 abilita sul mastro di registrazione l'uscita del buffer dalla rete di calcolo del codice ciclico CRCF0.

Il codice ciclico è calcolato nella maniera già vista nel caso di ricerca identificatore, l'unica variazione è data dall'ingresso alla rete di calcolo che deve essere quello relativo ai byte provenienti da U. C. e prelevati per l'appunto da FSC70. Durante COCI0, CRCF0 controllá il segnale di registrazione INREN.

COCI0 controlla la caduta di NREM0, di SEOK0, di AREA0 e di SCAC0. SEOK0 a sua volta con un ritardo di 8 µsec determina la caduta di ORRE0.

Nasce così di nuovo ORLE0 per il riconoscimento eventuale di un prossimo identificatore.

- Calcolo dei settori registrati e segnalazione di ultimo settore

Ad ogni REAM0 nasce un'interruzione per funzione di governo, come si è già visto nel caso di lettura ad ogni ritrovamento di una chiave di blocco che precedeva il settore da leggere. L'U. C. conta i settori registrati, aggiorna gli identificatori dei settori da registrare successivamente ed eventualmente emette la segnalazione di ultimo settore settando USET0.

- Logica di fine registrazione di tutti i settori

Come si è già visto per la lettura la caduta di SEOK0 in concomitanza con USET0 fa nascere una interruzione a livello 3A per fine.

Gli effetti di questa interruzione e i lavori svolti sono del tutto identici a quelli visti già nel caso di lettura.

TEMPORIZZAZIONE

Ricerca Identificatore e Registra Campo Dati (Parte iniziale)

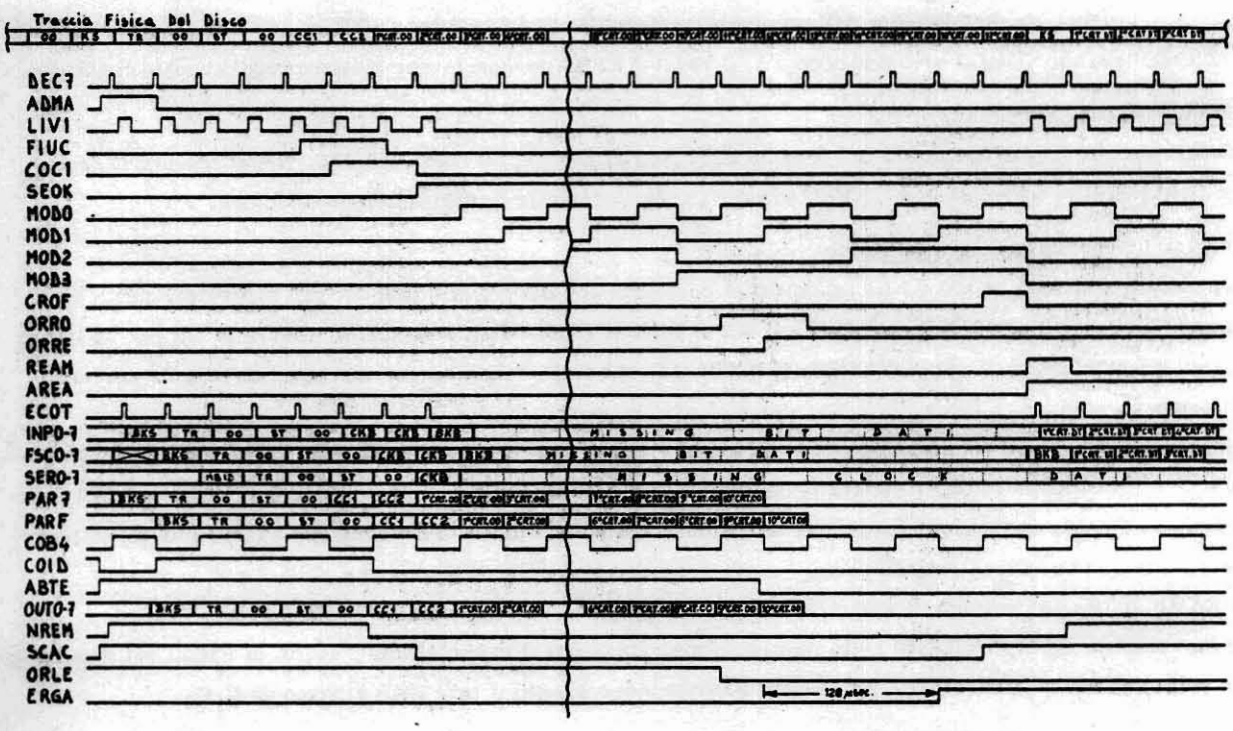


FIG. 18

TEMPORIZZAZIONE
Registra Campo Dati (Parte finale)

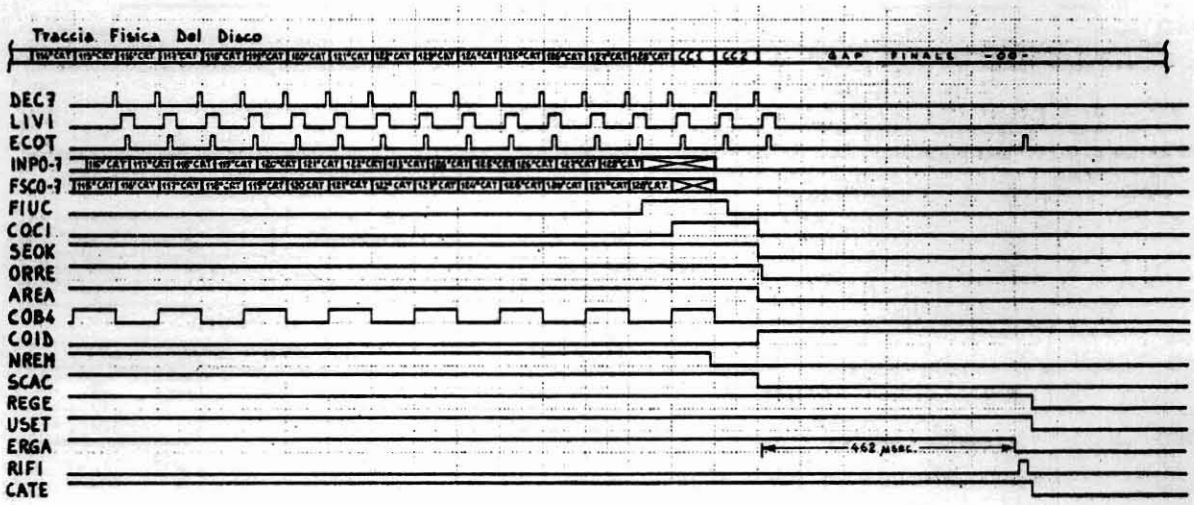


FIG. 19

- LOGICA DELLA VERIFICA -

La verifica è del tutto simile alla lettura, le uniche differenze si verificano alle interruzioni a livello 1 in cui l'U. C. anzichè introdurre i byte letti emette dei byte qualsiasi che non verranno considerati dal FLODI. Le interruzioni servono solo a controllare ECOF0 e quindi la temporizzazione di codice ciclico. Il comando che viene emesso da U. C. alla fine della conta tempo posiziona ad 1 il segnale SPEC0 sul buffer di staticizzazione dei comandi.

- Logica delle interruzioni a livello 1

Le interruzioni a livello 1 nascono col set di SCAC0 che avviene all'inizio di ADMA0.

Come si può vedere dalla temporizzazione di fig. 20, le interruzioni nascono a partire dallo stesso momento in cui partono nella registrazione e poichè il microprogramma svolto dall'U. C. a livello 1 è lo stesso della registrazione,come già in quel caso ECOF0 viene emesso con la $129^{a}/257^{a}$ interruzione e viene richiamato con la $131^{a}/259^{a}$ interruzione.

ECOF0 controlla COCI0 che abilita il controllo del codice ciclico come nel caso della lettura.

FORME D'ONDA DELLA VERIFICA

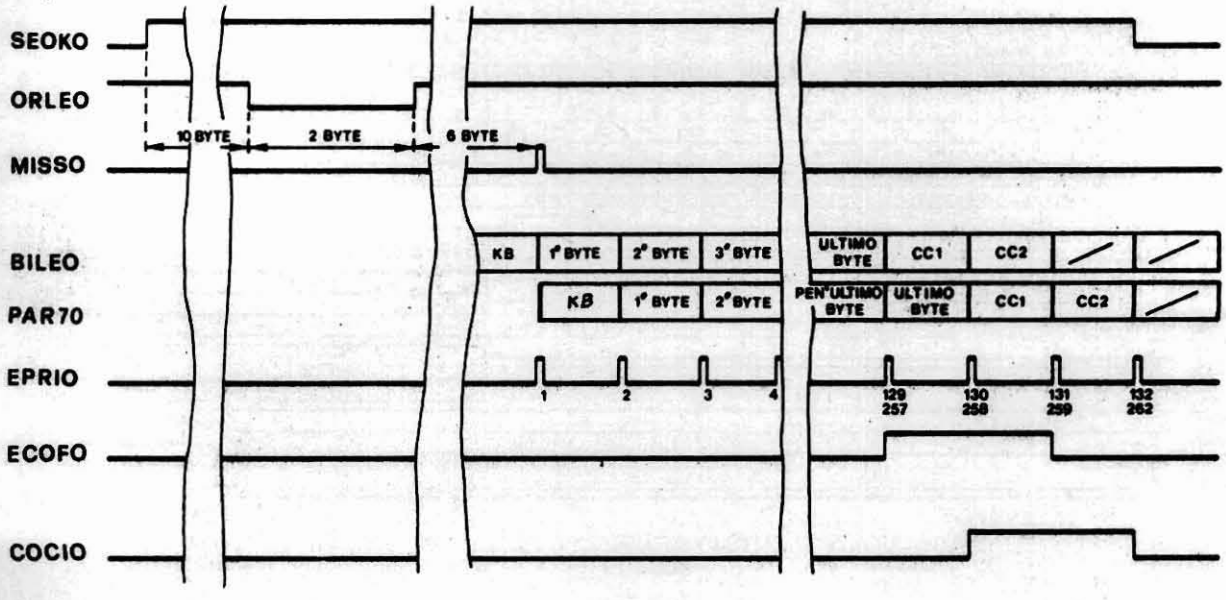


FIG. 20

- LOGICA DELLA SCANSIONE -

- Ultima interruzione per conta tempo ed emissione dei comandi da parte dell'U. C.

Sulla decima interruzione per funzione di governo vengono emessi due comandi sul bus dati ECD0 ÷ 7 accompagnati dagli strobe ECOT ed ECOC.

Il primo comando, essendo ancora PRIC0 a zero, viene staticizzato sul buffer dei comandi e posiziona rispettivamente SPEC0, SCAN0 e REG0 come indicato in tabella 6.

Il 2° comando trovando PRIC0 ad 1 viene staticizzato invece nel buffer MAS0 ÷ MAS7 (DISL005); esso é il complemento a FF 16 della lunghezza della maschera alla quale l'U. C., in precedenza, ha già sommato 1.

STI0	SCAN0	SPEC0	REG0
SCAN ⩾	1	1	0
SCAN ⩽	1	0	1
SCAN =	1	0	0

Tab. 6

- Conteggio lunghezza delle maschere e del numero di maschere confrontate

Riferirsi a figura 21

Durante ADMAN viene abilitata la carica parallela del contatore MOD0 ÷ 7 (DISL 005. M3 e M4) che riceve il contenuto del buffer MAS0 ÷ 7.

Dalla caduta di ADMAN ad ogni DEC7 il contatore conta avanti di 1; quando raggiunge la configurazione $[FF]_{16}$, il segnale MODU0 di fine conta riabilita la carica parallela col contenuto di MAS0 ÷ 7.

Il segnale MODU0 controlla il F. F. MAMO (DISL 005. M8) che si setta con DEC70 e si resetta poco dopo. MAM0 è il segnale che segnala la fine di una maschera sulla quale non è verificata la scansione e fa avanzare il contatore NUM0 ÷ 7 (DISL 001. G7 e G8).

Il contatore NUM0 ÷ 7 viene resettato all'inizio di ogni settore dal segnale MISS0. A scansione verificata (SCOK0 = 1) viene bloccata la conta su MOD0 ÷ 7 e quindi venendo a mancare MAM0 anche il contatore NUM0 ÷ 7 rimane fermo alla posizione raggiunta, memorizzando il numero di maschere confrontato prima che venisse verificata la scansione.

- Scambio dati con U. C. e decodifica dei byte $[FF]_{16}$.

SCAN0 abilita il set di SCAC0 (DISL 006. T6) con l'inizio di ADMA0 e SCAC0 a sua volta, come sempre, abilita le interruzioni a livello 1. (figura 21).

Le interruzioni termineranno con la caduta di COCI0 che avverrà con la $132^a/260^a$ interruzione come si può vedere in figura 22.

COCI0 come già in lettura abiliterà il controllo del codice ciclico.

I dati inviati da U. C., come già si è visto in registrazione, passano sul buffer INP0 ÷ 7 (DISL 004. C6 e C7) e quindi su FSC7. I dati emessi su INP0 ÷ 7 vengono decodificati ed i byte $[FF]_{16}$ controllano BLAN0 che sta in set per il byte successivo come risulta in fig. 21.

BLAN0 inoltre sta ad 1, forzato da BLASN (DISL 005. R7) finchè su FSC7 non è disponibile il primo byte da confrontare.

FORME D'ONDA D'ONDA D'INIZIO SCANSIONE

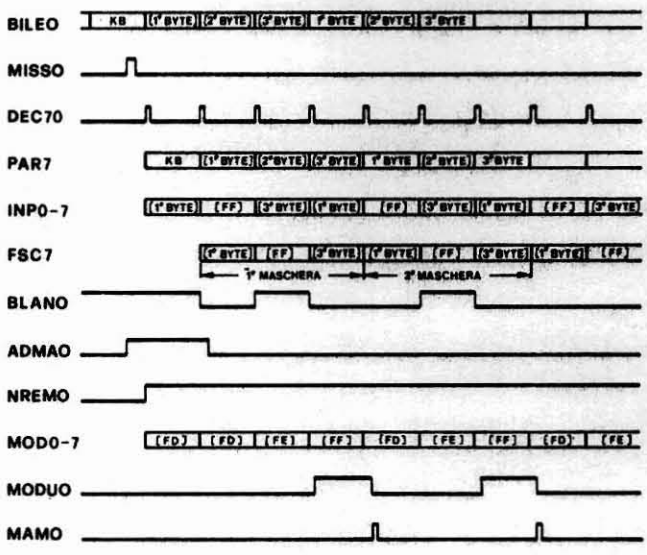


FIG. 21

- Confronto dei dati

Si è già detto che la scansione verificata è segnalata da SCOK0 in set e se la scansione è verificata su "maggiore" o "minore" viene posizionato anche DIVE0.

DIVE0 (DISL 005. R9) è messo a zero quando ancora SEOK0 è a zero e quindi viene rimesso a zero, nell'eventualità che si setti, alla fine di ogni maschera da MAM0.

Durante il confronto di ogni singola maschera DIVE0 è messo in set dai primi (più significativi) due bit diversi che compaiono su FSC7 e PAR7.

Il set di DIVE0 è bloccato da BLAN0 ad 1.

FORME D'ONDA DI FINE SETTORE

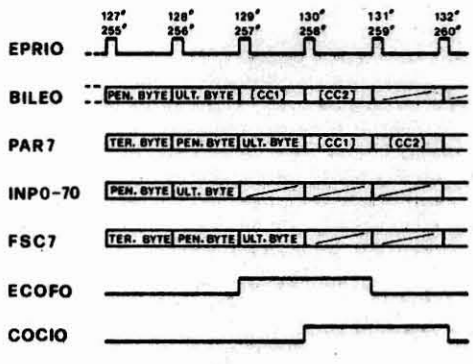


FIG. 22

SCOK0 può venire settato in tre modi diversi:

1 - Byte letti uguali alla maschera.
Alla fine di una maschera, seDIVE0 è rimasto in reset, SCO3N (DISL 005. G6) setta SCOK0.

2 - Byte letti > della maschera di U. C.
Se dei due bit che causano il set di DIVE0, quello su PAR7 è ad 1 e quello su FSC7 a zero, significa che il bit letto è maggiore di quello emesso da U.C. Poichè DIVE0 fino a quel momento era ancora a zero, significa che i bit in questione sono anche i più significativi dei bit diversi e che quindi tutto il campo letto è maggiore della maschera di U. C. (vedi figura 23). In questo caso il gate SCOIN (DISL 005. E6) setta SCOK0 se SPEC0 è ad 1.

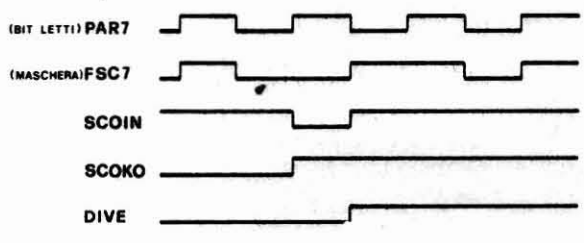


FIG. 23

3 - Campo letto < della maschera di U. C.

Il set di SCOK0 è dato da una situazione esattamente opposta a quella del caso 2, dal gate SCO2N (DISL 005. F6).

- Fine dell'operazione di scansione

La scansione può finire o perchè si è verificata in un settore qualsiasi, nel quale caso USET0 è forzato in set da SCOK0, o perchè sono finiti i settori su cui si doveva verificare la scansione, nel qual caso USET0 viene settato come si è già visto in lettura o in registrazione.

USET0 fa nascere l'interruzione per fine, come già visto nelle altre operazioni.

Si ricorda che la seconda volta in cui l'U. C. introduce il bus dati EPD0 ÷ 7 riceve il contenuto del contatore NUM0 ÷ 7 che contiene il numero di maschere confrontate prima della maschera su cui si è verificata la scansione.

NOTE DI PRINCIPIO SUL PROGRAMMA DI IMMATRICOLAZIONE

Un programma di immatricolazione serve per predisporre un disco nuovo di fabbrica per l'uso corrente su un sistema.

Il programma prevede l'introduzione in memoria di una serie di byte che serviranno a registrare gli identificatori di tutti i settori e di tutte le tracce a seconda del formato che si vuole registrare.

Coi bit dell'identificatore devono essere memorizzati anche i byte di codice ciclico delle diverse chiavi di settore poiché il FLODI non sarà in grado di fornirli.

Il programma di immatricolazione prevederà un ritorno della testina in traccia "00" e quindi una serie di 77 operazioni di posizionamento o registra traccia sulle 77 diverse tracce del disco. In ogni traccia viene registrato ciò che è indicato in figura 24

Dopo aver predisposto gli identificatori ed i gap si comanda la registrazione di dati standard su tutti i settori di tutte le tracce.

In questo modo vengono registrati, in ogni settore:

- 6 byte $[00]_{16}$ alla fine di ogni gap separatore
- chiave di blocco
- 128/256 byte
- codice ciclico di controllo

La registrazione di tutti i blocchi sarà seguita da una verifica su tutti i blocchi e l'U. C. potrà così sapere quali tracce siano eventualmente difettose.

Gli identificatori delle tracce difettose saranno registrati con 4 byte $[FF]_{16}$ con l'operazione di registra traccia difettosa.

Sulla traccia "00" sarà invece registrato l'indirizzo delle tracce risultate difettose per rendere disponibile l'informazione al momento della utilizzazione.

Si ricorda che attualmente esistono 3 tipi di formato traccia:

1) con 15 settori per traccia con separatori di byte $[FF]_{16}$ e $[00]_{16}$
2) con 26 settori per traccia e separatori di byte $[00]_{16}$
3) con 26 settori per traccia e separatori di byte $[FF]_{16}$ e $[00]_{16}$

La logica del FLODI permette di registrare i tre formati senza modifiche di hardware.

I tre diversi formati sono ottenuti con diverse predisposizioni in memoria fatte col programma di immatricolazione.

L'operazione comincia come una normale registrazione con un posizionamento e una conta tempo.

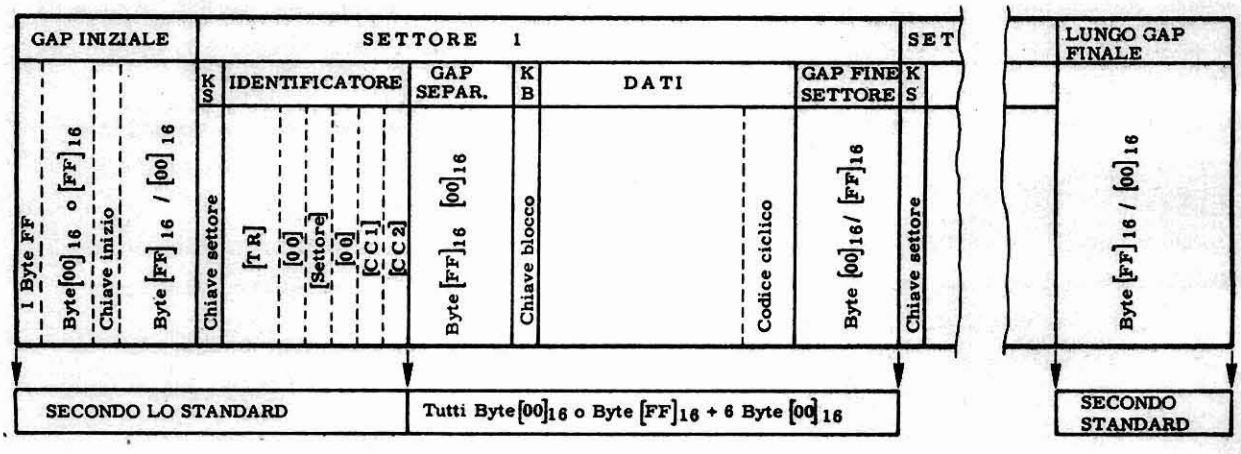


FIG. 24

- LOGICA DELL'OPERAZIONE "REGISTRA TRACCIA" -

- Ultima interruzione per conta tempo

Il primo comando emesso da U. C. posiziona SPEC0 e REG00 ad 1 e SCAN0 a zero che generano RETR0 (DISL 002. T7).

Il primo comando, come si è già visto, setta PRIC0 in modo che il secondo comando generi INCO0 (DISL 005. A2).

Nel caso di registra traccia INCO0 causa una interruzione a livello 1 con RILSN (DISL 006. M5).

Durante l'interruzione viene predisposto su INP0 + 7 un byte $[00]_{16}$ o un byte $[FF]_{16}$ che verrà registrato come secondo byte del gap iniziale.

Si noti che il byte emesso, mancando interruzioni a livello 1, rimane bloccato su INP0 + 7.

- Avvio delle registrazioni

Riferirsi alle figure 25 e 26

Il segnale INDEA proveniente da FLODISC genera INDO0 (DISL 004. G1) come si è già visto in lettura o in registrazione.

Durante l'operazione di registra traccia INDO0 genera INABN (DISI 004. G3) che setta ORRE0 e SCAC0.

ORRE0 controlla il mastro di registrazione INRE0 (DISL 008. E9) ed abilita la registrazione dei clock.

- Registrazione del 1° byte del gap iniziale

Prima che venga emesso il segnale di conta tempo, il F. F. BIAN0 (DISL 005) viene messo in set dal segnale BLASN (DISL 005. R7). BLAN0 forza a zero REBLN (DISL 004. S5) che abilita la registrazione di bit 1 sul mastro INREN (DISL 008. E8) come è indicato in figura 25

- Logica di registrazione del secondo byte del gap iniziale

Il primo DEC7 conferma BLAN0 ad 1 o lo pone a zero a seconda della configurazione di INP0 + 7.

Il buffer INP0 + 7 contiene il byte che gli è stato dato durante la interruzione a livello 1 eseguita durante l'ultima interruzione di conta tempo.

La figura 25 riporta il caso in cui BLAN0 abilita la registrazione di un byte $[00]_{16}$ e la figura 26 riporta il caso di un byte $[FF]_{16}$.

- Registrazione della chiave di traccia

Durante la 44ª interruzione a livello 1 l'U. C. emette un byte $[FC]_{16}$ e durante la 45ª un byte $[D7]_{16}$.

Il byte $[FC]_{16}$ entra su INP0 + 7 quindi su FSC7 e SER7. Il byte $[D7]_{16}$ entra su INP0 + 7, su FSC7 e su SER7 solo se il byte emesso da U. C. sulla 46ª interruzione è un byte $[00]_{16}$ altrimenti è bloccato da BLAN0 come è indicato in figura 25.

Quando il 1° bit del byte $[FC]_{16}$ esce da SER7, viene settato il F. F. REAM0 (DISL 004. C4) che sta in set per un byte; nell'eventualità che il byte $[D7]_{16}$ passi sul SER7 il F. F. NREM0 gli impedisce di far ripartire REAM0.

REAM0, come già in registrazione, controlla la scrittura dei clock in funzione di SER7 (DISL 008. C5).

Con REAM0 si setta AREA0 che abilita l'uscita di FSC7 sul nastro. AREA0 è resettato con l'emissione di un ECOF0 da parte dell'U. C.

Gli altri bit della chiave di traccia sono registrati col controllo di REBLN.

- Registrazione dell'identificatore

La logica è del tutto identica a quella della registrazione della chiave di traccia, ci affidiamo perciò alla temporizzazione di figura 25 e 26 per risolvere eventuali dubbi.

- Controllo dei settori registrati e segnalazione ultimo settore

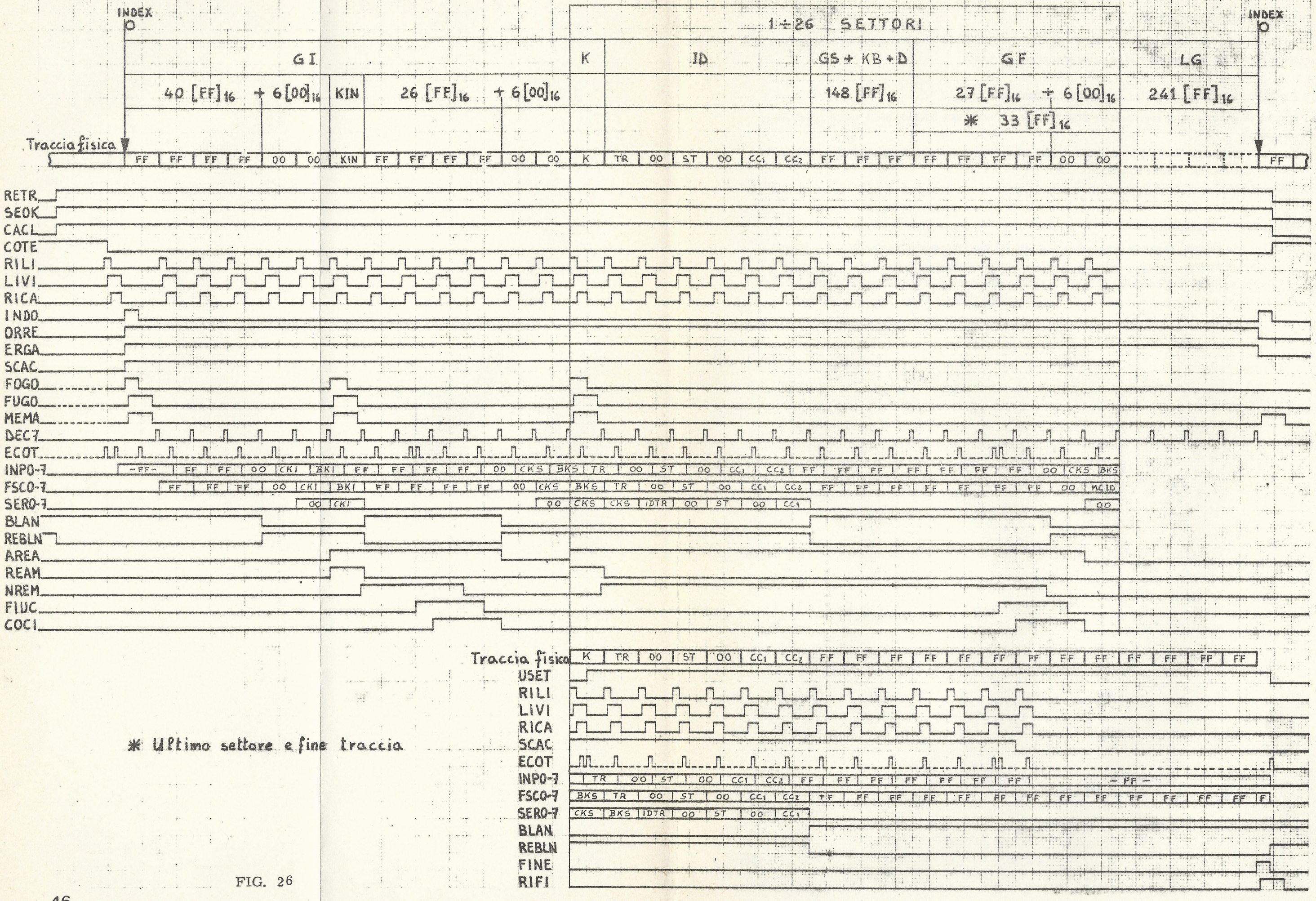
REAM0 setta come già in registrazione FOGO0 e quindi genera delle interruzioni a livello 3 per funzioni di governo.
L'U. C. conteggia i settori registrati e con l'ultimo settore emette la segnalazione per il set di USET0 che alla fine di COGO resetta SCAC0.
Vengono così troncate le richieste di interruzioni a livello 1.

- Registrazione del gap finale

L'ultimo byte emesso, dopo aver emesso l'ultimo byte dal gap separatore è un byte $[FF]_{16}$ o $[00]_{16}$.
L'ultimo byte emesso, venendo a mancare altre interruzioni, rimane sul buffer INP0 ÷ 7 e controlla continuamente REBLN che per tutta la durata di ORRE0 farà registrare $[00]_{16}$ oppure $[FF]_{16}$.
ORRE0 è fatto cadere con il segnale di INDEX che trova USET0 in set mentre viene avviata l'interruzione di fine.

TEMPORIZZAZIONE

IMMATRICOLAZIONE TRACCIA $[FF]_{16}$

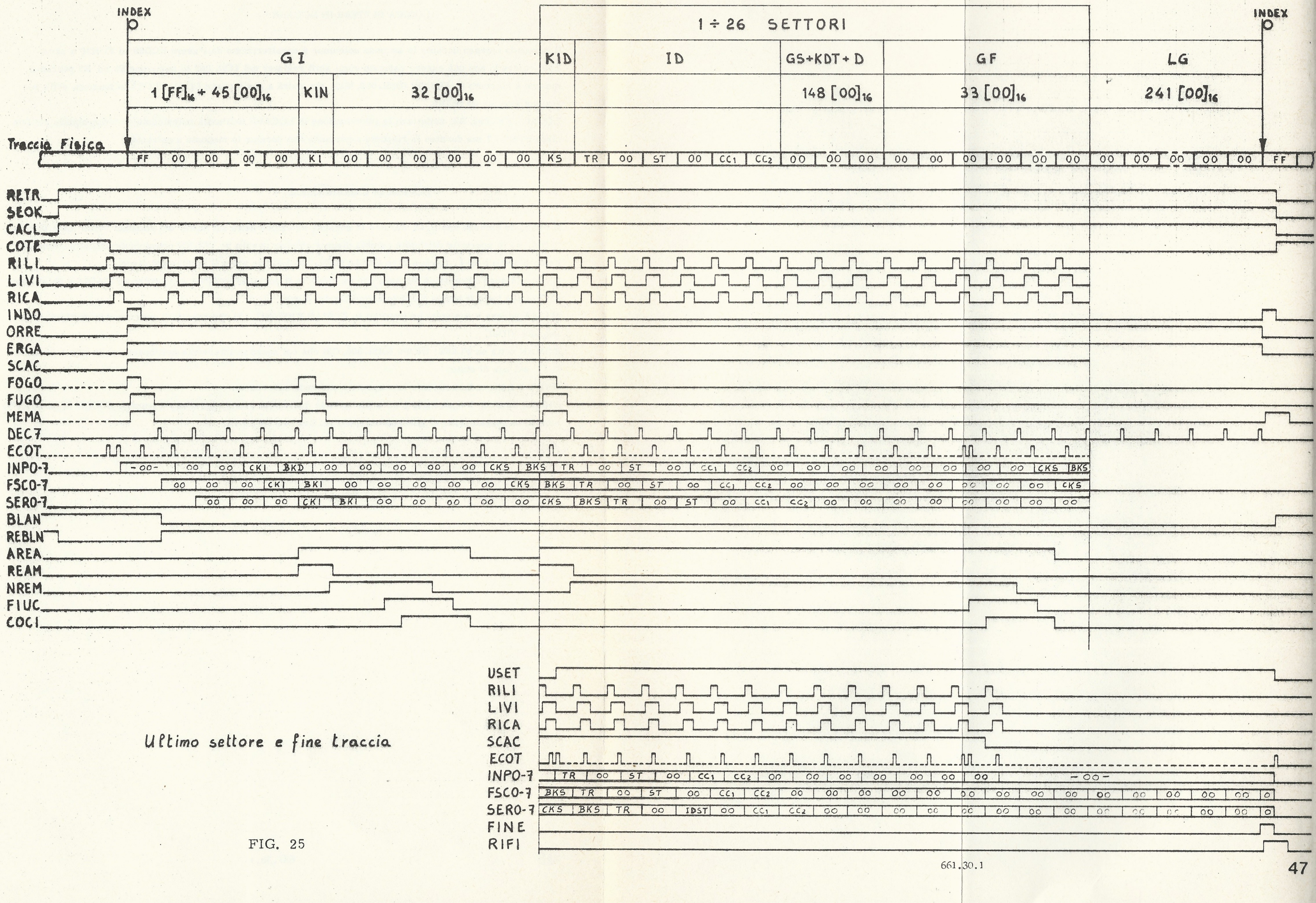


* Ultimo settore e fine traccia

FIG. 26

TEMPORIZZAZIONE

IMMATRICOLAZIONE TRACCIA $[00]_{16}$



Ultimo settore e fine traccia

FIG. 25

661.15.1

661.30.1

- LOGICA DI "PONI IN LOCALE" -

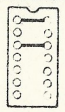
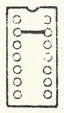
Il comando emesso durante la seconda selezione è caratterizzato dall'avere ECD60 ed ECD70 a zero.
Tale comando non può essere caricato sugli shift register del DISL 002 in posizione E8 ed E9 perché il mastro è bloccato da COLON (DISL 002. M1), ma andrà a posizionare LOC1 o LOC2 in funzione della posizione assunta da SEDI0.
LOCA0 (DISL 002. R2) sintetizza la informazione di LOCALE dell'unità selezionata ed è disponibile sul bus stato EPD0 ÷ 7 sia durante la risposta a comando che durante la risposta a selezione.

- LOGICA DI CAMBIA DISCO -

Il comando emesso con U. C. durante la seconda selezione setta sul buffer dei comandi VIDI0 (DISL 002. E9).
VIDI0, se lo sportello dell'unità cambia disco è chiuso, ovvero se VOLNN proveniente dall'attuazione del cambio disco è ad 1, porta a zero CADIA (DISL 009. L4) che avvia il cambio del disco.
L'alimentatore automatico dichiara che ha ricevuto il comando inviando un impulso su MADIA e abbassando CICOA per tutto il tempo relativo al cambio del disco.
MADIA genera a sua volta un impulso su ECMAN (DISL 002. F7) che resetta il governo e in particolare il segnale VIDI0.
ECMAN setta su FLODI INOP0 (DISL 008. C2) che col senso di intervento dell'operatore in atto è inviato a U. C. sul bus di stato.
INOP0 è fatto cadere durante la risposta a comando successiva.
CICOA, invece, è mantenuto a zero per tutta la durata del cambio del disco e tramite GOCON (DISL 007. P7) emette la segnalazione di governo occupato sul bus dati EPD0 ÷ 7.

TABELLE DA RITAGLIARE

TABELLA DEI CODICI DI SELEZIONE E DEL BUS NOME

DRIVER	Nome software	Bus selezione 7N	6N	5N	4N	3N	2N	1N	0N	Bus nome 7N	6N	5N	4N	3N	2N	1N	0N	PONTICELLI
1	C.0	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	
2	C.1	0	0	0	1	1	1	1	1									
1	D.0	1	0	0	1	0	1	1	1	1	0	0	1	0	1	1	1	
2	D.1	0	0	0	1	0	1	1	1									

Tab. 2

Tab. 3

BUS TIPO

Liv. interruz.	Descrizione della interruzione	BUS TIPO SU EPT0N 7N	6N	5N	4N	3N	2N	1N	0N
1	Scambio dati	1	1	1	1	1	1	1	*
3A	Fine comando classe C	1	1	1	1	1	0	1	1
3A	Funzione di governo	1	1	1	1	0	1	1	1
3B	Risposta a comando	1	1	1	1	1	1	0	0
3B	Risposta a selezione	1	1	1	1	1	1	1	0

* Viene emesso **0 se ERORØ = 1, ovvero** se c'è errore di codice ciclico o non è ritrovato l'identificatore voluto.

SIGNIFICATO DEI BIT EPD0 + 7 NEI VARI CASI

BIT	SCAMBIO CARATTERI	RISPOSTA A SELEZ.	RISPOSTA A COMANDO	FUNZIONE GOVERNO	FINE COMANDO CLASSE "C" 1° INPUT (PRIC0=0)	2° INPUT (PRIC0=1)
EPD00	OUT00			IND0 : INDEX	IND0 : INDEX	NUM00
EPD10	OUT10	GOC0 - OR logico di: - Via carrello - Testina abbassata - Cambio disco	GOC0 - OR logico di: - Via carrello - Testina abbassata - Cambio disco	COTE: Conta tempo	COTE: Conta tempo	NUM10
EPD20	OUT20	GOC0 - OR logico di: - Via carrello - testina abbassata - Cambio disco	GOC0 - OR logico di: - Via carrello - Testina abbassata - Cambia disco	TEVE: Scansione verificata su diverso	TEVE: Scansione verificata su diverso	NUM20
EPD30	OUT30	INOP: Intervento operatore	INOP: Intervento operatore	SCOK: Scansione verificata	SCOK: Scansione verificata	NUM30
EPD40	OUT40	LOCA: DRIVER selezionato in locale	LOCA: DRIVER selezionato in locale			NUM40
EPD50	OUT50	MADI: Macchina disponibile	MADI: Macchina disponibile	MADI: Macchina disponibile	MADI: Macchina disponibile	NUM50
EPD60	OUT60	PIZE: Carrello in traccia 00	PIZE: Carrello in traccia 00	PIFU = PIZE: Carrello in traccia 00	PIFU = ERR0: Errore in codice ciclico	NUM60
EPD70	OUT70	DIPE: Disco non presente o sportello aperto	DICA: Disco non inserito o sportello aperto	ERAS: Blocco deleted	ERAS: Blocco deleted	NUM70

Tab. 4

COMANDI EMESSI E SEGNALI PREDISPOSTI SU FLODI

	OPERAZIONE ST10 O SI0	CODICE SOFTWARE	CLASSE COMANDO	RISPOSTA ALLA 2ª SELEZIONE 1° comando PRIC0 = 0	2° comando PRIC0 = 1	ULTIMA FUNZIONE DI POSIZIONAMENTO CARRELLO	ULTIMA FUNZIONE DI GOVERNO PER CONTA TEMPO O PER CONTA SETTORI 1° Comando PRIC0 = 0	2° Comando PRIC0 = 1	INTERRUZIONE DI FINE COMANDO DI CLASSE "C" O FINE CONTA INDEX
1	POSIZIONAMENTO SU UNA TRACCIA QUALSIASI	4.4	B	0100 x 010 ECD10-SPEC0 ECD30-DIRI0 ECD40-CLAB0 NOTA 1	0101 x 010 ECD10-SPEC0 ECD30-DIRI0 ECD40-VIRI0 ECD60-CLAB0	11 00 00 00 ECD60-CLAB0 ECD70-CATE0	00000000		
2	POSIZIONAMENTO SU TRACCIA "00"	4.8	B	NOTA 2 come sopra	come sopra	come sopra	come sopra		
3	LOCALE			00 000000	0000 0000				
4	CAMBIA DISCO			0110 x 010 ECD5 - VIDI0 ECD6 - CLAB0	0110 x 010 ECD5 - VIDI0 ECD6 - CLAB0				
5	POSIZIONAMENTO E LETTURA	C.0	C	0100 x 000 ECD30-DIRI0 ECD60-CLAB0 NOTA 1	0101 x 000 ECD30-DIRI0 ECD40-VIRI0 ECD60-CLAB0	10 00 0000 ECD7 - CATE0	1000 0000 ECD7 - CATE0	0000 0000 MAS0 + 7 [00] 16	0000 0000
6	POSIZIONAMENTO E REGISTRAZIONE BLOCCHI DELETED O UNDELETED	8.1 8.0	C C	come sopra NOTA 1	come sopra	10 00 0001 ECD70 - CATE0 ECD00 - REGO0	10 00 0001 ECD70 - CATE0 ECD00 - REGO0	0000 0000 MAS0 + 7 [00] 16	0000 0000
7	POSIZIONAMENTO E VERIFICA	8.2	C	come sopra NOTA 1	come sopra	1000 0010 ECD10 - SPEC0 ECD70 - CATE0	1000 0010 ECD10 - SPEC0 ECD70 - CATE0	0000 0000 MAS0 + 7 [00] 16	0 000 0000
8	POSIZIONAMENTO E SCANSIONE SU →	8.6	C	come sopra NOTA 1	come sopra	1000 0110 ECD10 - SPEC0 ECD20 - SCAN0 ECD70 - CATE0	1000 0110 ECD10 - SPEC0 ECD20 - SCAN0 ECD70 - CATE0	xxxx xxxx MAS0 + 7 [xxxx xxxx]	0000 0000
9	POSIZIONAMENTO E SCANSIONE SU ←	8.4	C	come sopra NOTA 1	come sopra	10 000 100 ECD20 - SCAN0 ECD70 - CATE0	1000 0100 ECD20 - SCAN0 ECD70 - CATE0	xxxxxxxx MAS0 + 7 [xxxxxxxx]	0000 0000
10	POSIZIONAMENTO E SCANSIONE SU ↓	8.5	C	come sopra NOTA 1	come sopra	1000 01 01 ECD00 - REGO0 ECD20 - SCAN0 ECD70 - CATE0	1000 0101 ECD00 - REGO0 ECD20 - SCAN0 ECD70 - CATE0	xxxxxxxx MAS0 + 7 [xxxxxxxx]	0000 0000
11	POSIZIONAMENTO E REGISTRA TRACCIA	8.3	C	come sopra NOTA 1	come sopra	1000 0011 ECD00 - REGO0 ECD10 - SPEC0 ECD70 - CATE0	1000 0011 ECD00- REGO0 ECD10- SPEC0 ECD70 CATE0	0000 0000	0000 0000

NOTA 1 - Nel caso in cui la testina sia già posizionata il comando emesso è quello che normalmente viene emesso sulla ultima funzione di governo per posizionamento carrello.

NOTA 2 - Anche se la testina è già in traccia zero si dà il via al carrello e si ha un PAVAA

Tab. 5

SCHEMI A BLOCCHI

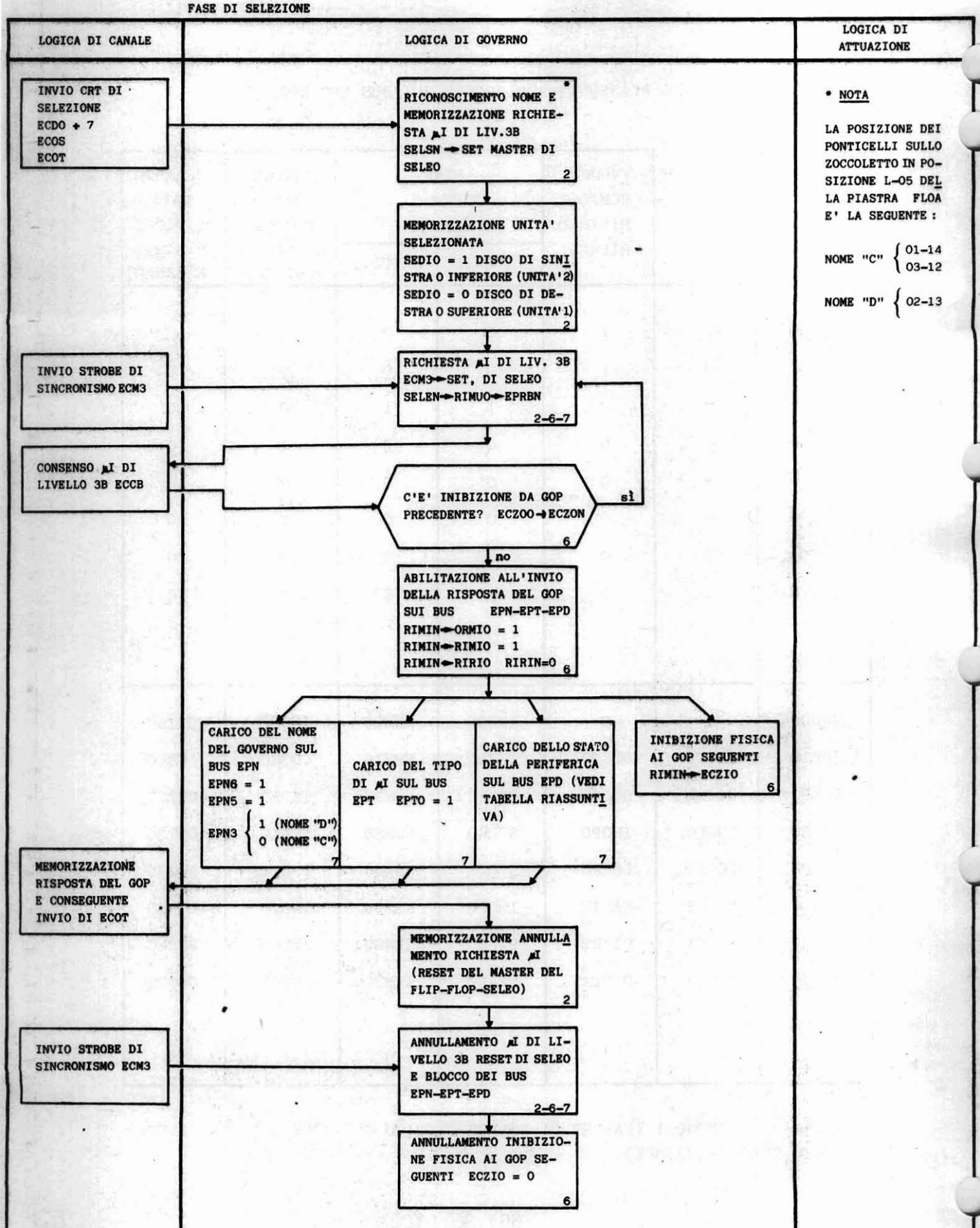
661 (02)

TABELLA RIASSUNTIVA RISPOSTE SUI BUS EPT EPD

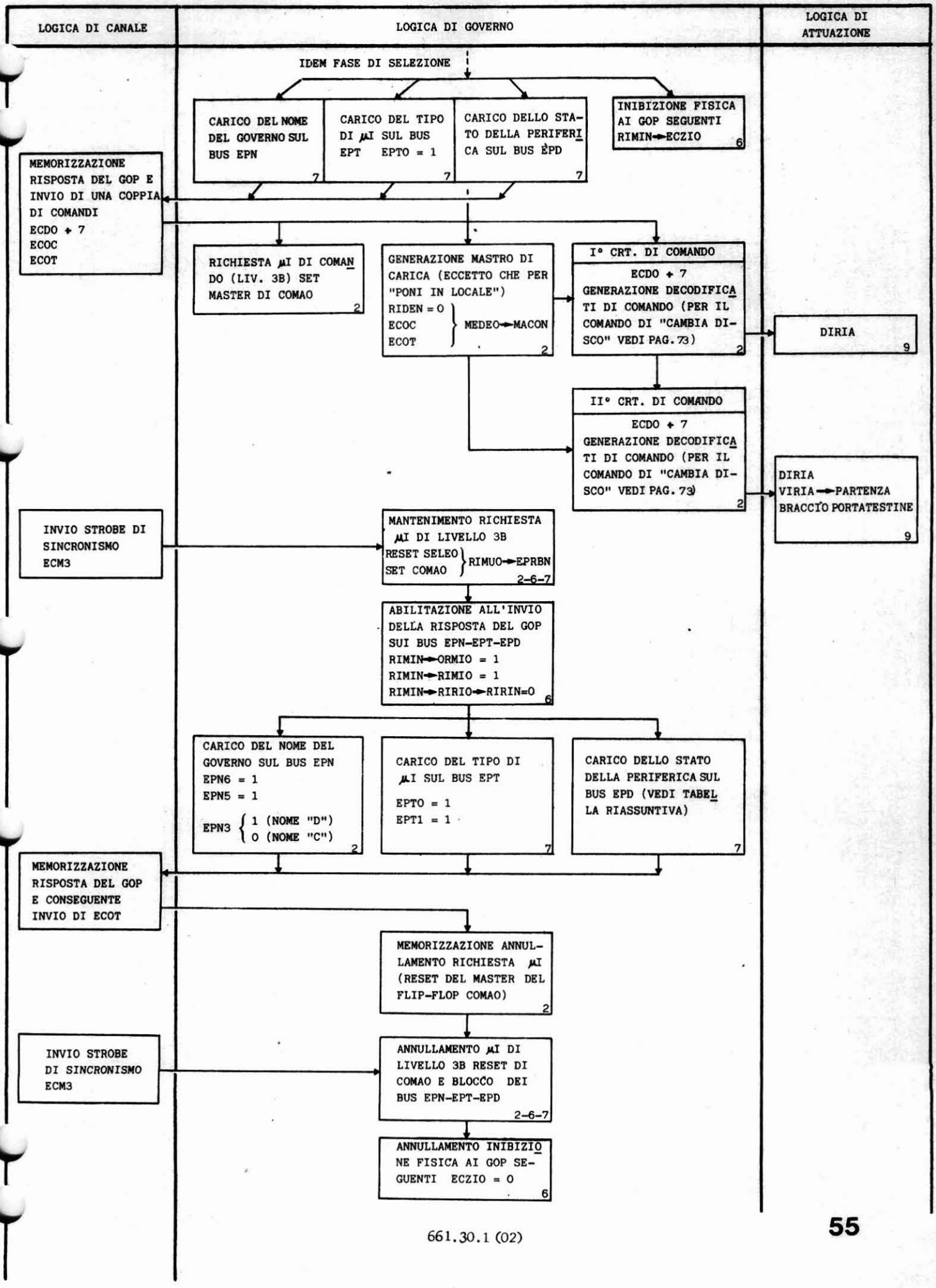
BIT	SELEZIONE SELEO=1 RIRIN=0 RICAN=1	COMANDO COMAO=1 RIRIN=0 RICAN=1	FINE RIFIO=1 MEMAN=0		FUNZIONI DI GOVERNO FUGOO=1 MEMAN=0 PRICO=0	SCAMBIO DATI LIVIO=1 RIRIN=0 RICAN=0
			PRICO=0	PRICO=1		
EPT00	1	1	0	0	0	0/1*
EPT10	0	1	0	0	0	0
EPT20	0	0	1	1	0	0
EPT30	0	0	0	0	1	0
EPT40	0	0	0	0	0	0
EPT50	0	0	0	0	0	0
EPT60	0	0	0	0	0	0
EPT70	0	0	0	0	0	0
EPD00	0	0	INDOO	NUMOO	INDOO	OUTOO
EPD10	GOCOO	GOCOO	COTEO	NUM10	COTEO	OUT10
EPD20	GOCOO	GOCOO	TEVEC	NUM20	TEVEO	OUT20
EPD30	INOPO	INOPO	SCOKO	NUM30	SCOKO	OUT30
EPD40	LOCAO	LOCAO	0	NUM40	0	OUT40
EPD50	MADIB	MADIB	INOPO	NUM50	INOPO	OUT50
EPD60	PIZE1	PIZE1	PIFUO	NUM60	PIFUO	OUT60
EPD70	DIPEA	DIPEA	ERASO	NUM70	ERASO	OUT70

* EPTOO=1 SE ERORO=1 (ERRORE DI CODICE CICLICO O ERRORE DURANTE IL CONFRONTO IDENTIFICATORE).

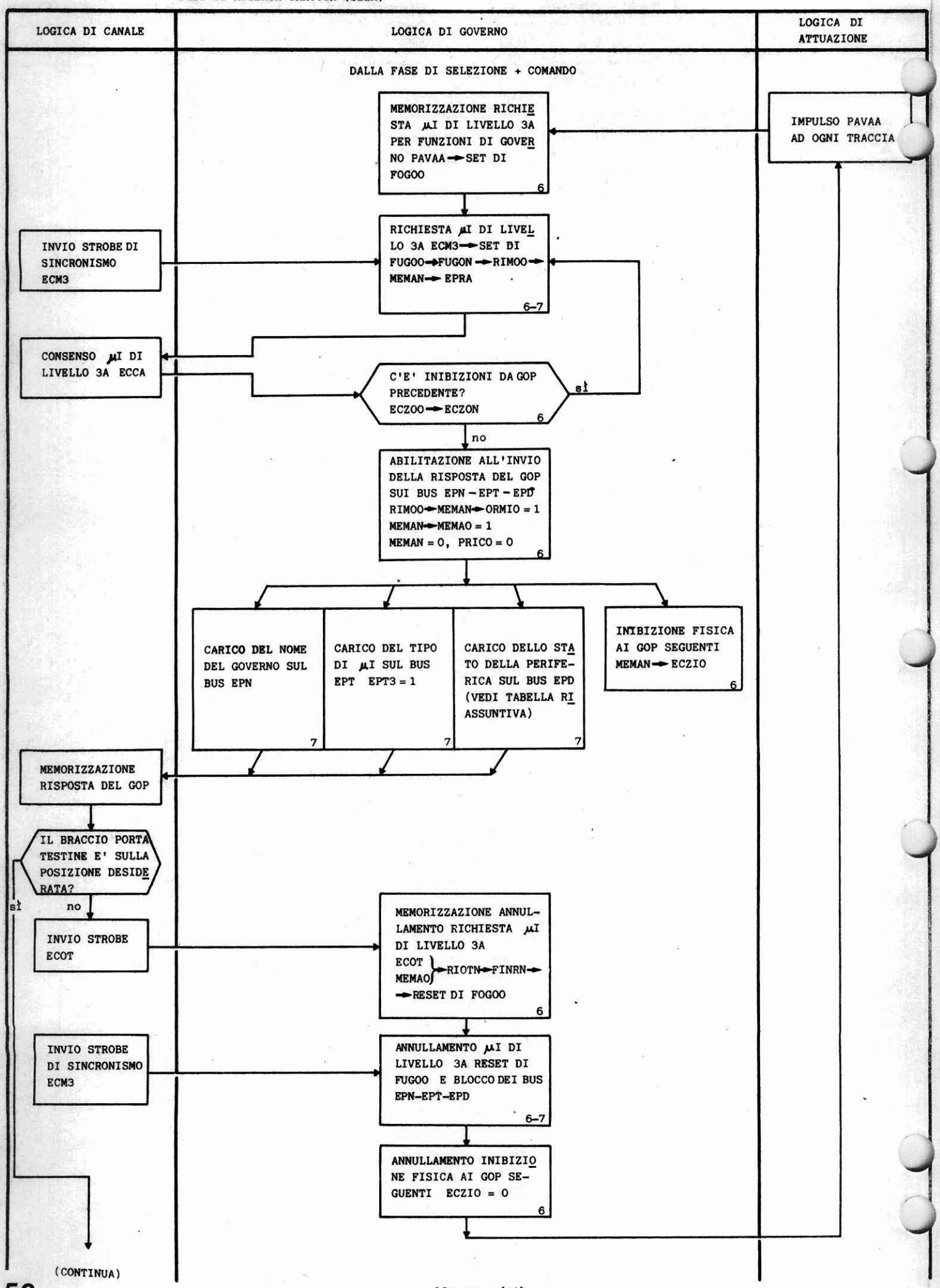
FASE DI SELEZIONE



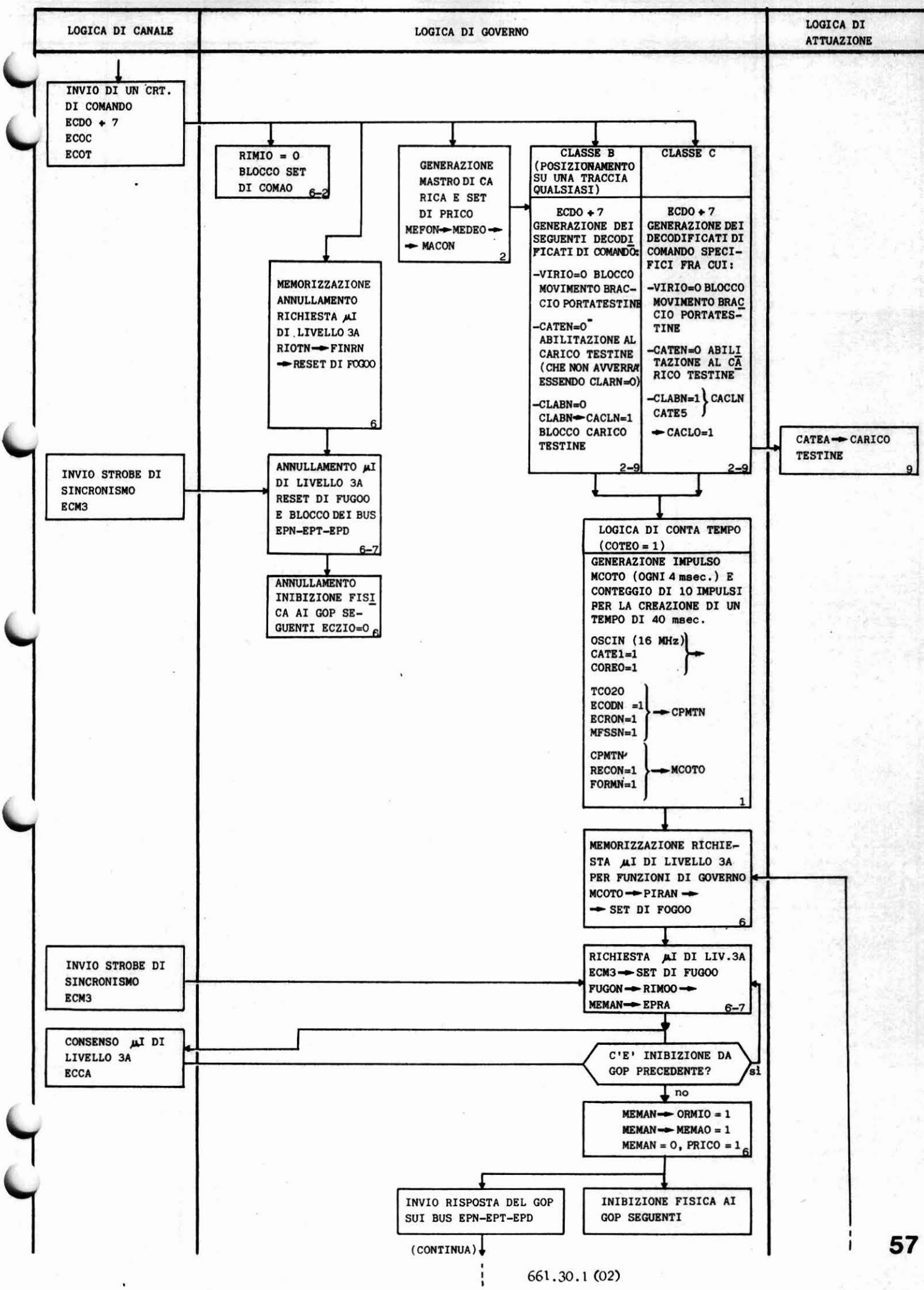
FASE DI SELEZIONE + COMANDO



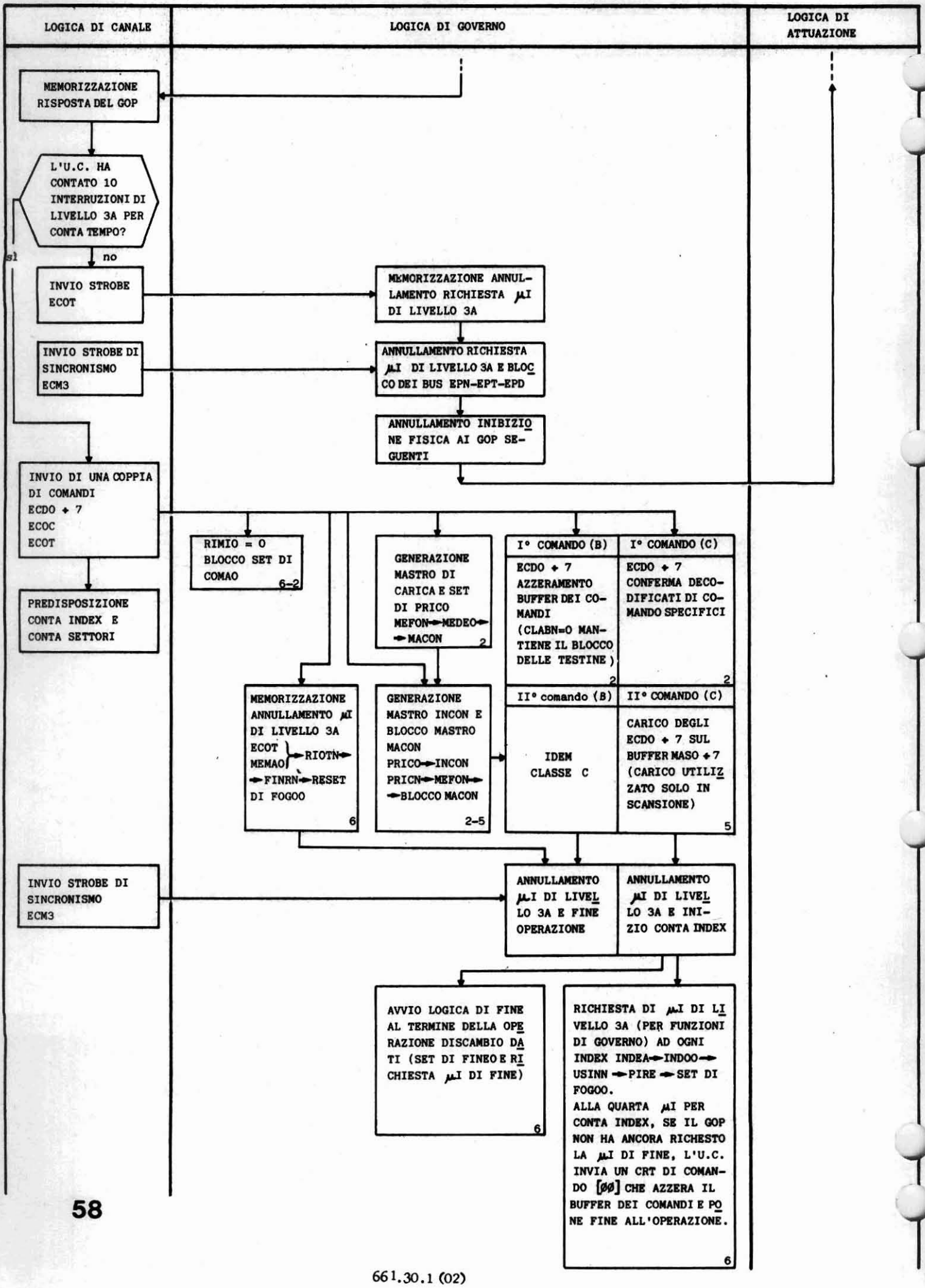
FASE DI RICERCA TRACCIA (SEEK)



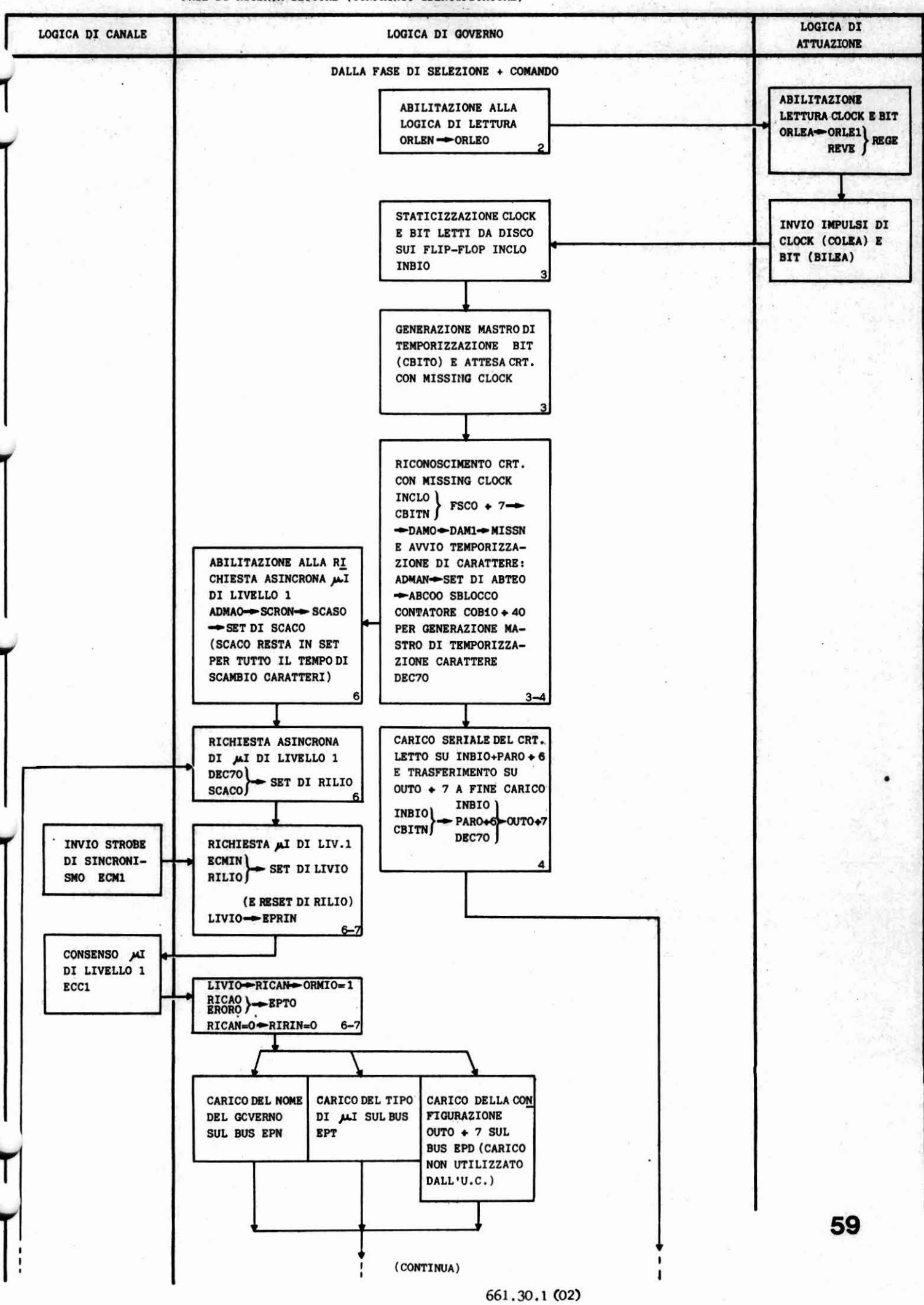
CONTINUAZIONE FASE DI RICERCA TRACCIA (SEEK)



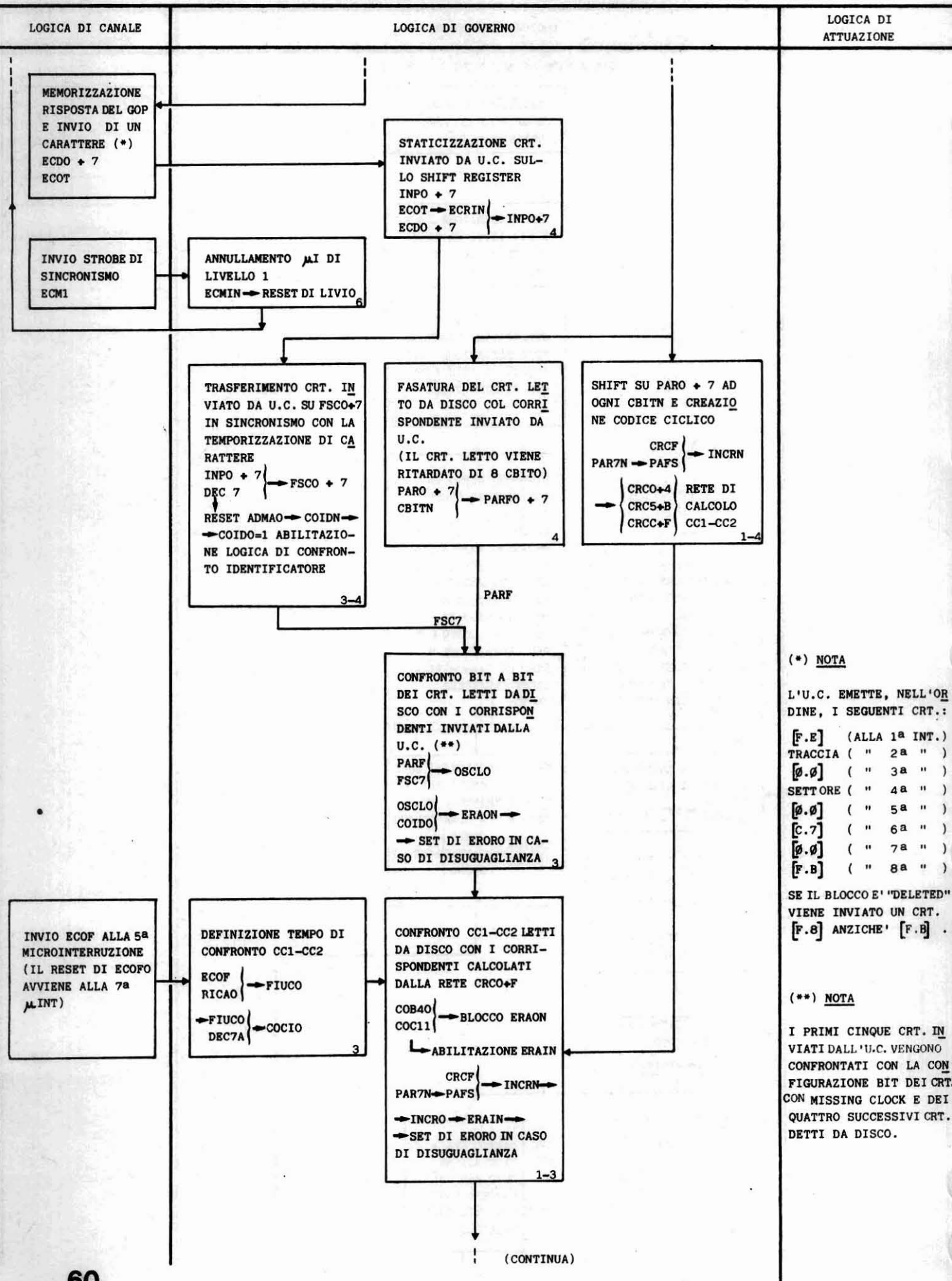
CONTINUAZIONE FASE DI RICERCA TRACCIA (SEEK)



FASE DI RICERCA SETTORE (CONFRONTO IDENTIFICATORE)



CONTINUAZIONE FASE DI RICERCA SETTORE (CONFRONTO IDENTIFICATORE)



(*) NOTA

L'U.C. EMETTE, NELL'ORDINE, I SEGUENTI CRT.:

[F.E] (ALLA 1ª INT.)
TRACCIA (" 2ª ")
[Ø.Ø] (" 3ª ")
SETTORE (" 4ª ")
[Ø.Ø] (" 5ª ")
[C.7] (" 6ª ")
[Ø.Ø] (" 7ª ")
[F.B] (" 8ª ")

SE IL BLOCCO E' "DELETED" VIENE INVIATO UN CRT. [F.8] ANZICHE' [F.B] .

(**) NOTA

I PRIMI CINQUE CRT. INVIATI DALL'U.C. VENGONO CONFRONTATI CON LA CONFIGURAZIONE BIT DEI CRT. CON MISSING CLOCK E DEI QUATTRO SUCCESSIVI CRT. DETTI DA DISCO.

CONTINUAZIONE FASE DI RICERCA SETTORE (CONFRONTO IDENTIFICATORE)

LOGICA DI CANALE	LOGICA DI GOVERNO	LOGICA DI ATTUAZIONE

MEMORIZZAZIONE ESITO CONFRONTO IDENTIFICATORE ALLA 8a MICRO-INTERRUZIONE DI LIVELLO 1

CARICO DELL'ESITO DEL CONFRONTO SUL BUS EPT PER INVIO ALL'U.C.
ERORO → EPTO

ERORO = 0 } CONFRONTO
EPTOO = 0 } O.K.

7

ATTESA SUCCESSIVO CRT. CON MISSING CLOCK PER NUOVO CONFRONTO IDENTIFICATORE

SEOKO = 1 ?

3

no

sì

ABILITAZIONE SHIFT AD ANELLO DEL REGISTRO SERO+7 (CONTENENTE IL CRT. [C.7] , OVVERO LA CONFIGURAZIONE CLOCK DELLA CHIAVE DI BLOCCO, CHE RESTA DISPONIBILE PER EVENTUALI SUCCESSIVE UTILIZZAZIONI)

SEOKO=1 → SEREO → → SELEZIONE INGRESSO SER7O

4

CARICO DI INPO+7 (OVE E' MEMORIZZATO L'ULTIMO CRT. INVIATO DA U.C., OVVERO LA CONFIGURAZIONE BIT DELLA CHIAVE DI BLOCCO [F.3] / [F.8]) SU FSCO+7 AD OGNI DEC7, E SHIFT DEGLI FSCO+7 AD OGNI CBITO.
DALL'USCITA FSC7 ESCE SERIALMENTE IL CRT. [F.3] / [F.8] CHE RESTA DISPONIBILE PER EVENTUALI SUCCESSIVE UTILIZZAZIONI

4

LETTURA CAMPO DATI

LOGICA DI CANALE | LOGICA DI GOVERNO | LOGICA DI ATTUAZIONE

DALLA FASE DI RICERCA SETTORE

REGOO = 0
SPECO = 0
SCANO = 0
2

RITROVAMENTO SETTORE CERCATO SEOKO = 1
3

BLOCCO ORLEO DURANTE IL TEMPO DI LETTURA DELL'11° E 12° CRT. DOPO IL CC2 (ZONA DI COMMUTAZIONI SPURIE DI REGISTRAZIONE)
SEOKO SBLOCCO CONTATORE MODO+7
(8) MOD3O SET
(10) MODON RESET } ORROO
ORRON=O→ORTE→ORLEN →ORLEO=0
(ORROO RESTA IN SET DURANTE IL TEMPO DI LETTURA DELL'11° E 12° CRT. DOPO IL CC2; DURANTE QUESTO TEMPO CBITO VIENE GENERATO DALLA TEMPORIZZAZIONE INTERNA-MOO4O- ANZICHE' DA COLEB)
2-4-5

ORLEO=O→ORLEA=1 BLOCCO DELLA LETTURA DELL'11° E 12° CRT.

RICONOSCIMENTO CRT. CON MISSING CLOCK (CHIAVE DI BLOCCO)
MISSN → SET DI ADMAO→ NASCITA DEC7 E SET DI NREMO AL PRIMO CRT. UTILE DEL CAMPO DATI (NREMO RESTA IN SET PER TUTTO IL TEMPO DI LETTURA DEI CRT. DEL CAMPO DATI COMPRESI CC1-CC2)
ADMAN → ABTEO → ABTEE →ABCOO → NASCITA DEC7
ADMAN→ ADREO } SET DI
DEC7 } NREMO
3-4

STATICIZZAZIONE CRT. CON MISSING CLOCK SU INBIO+PARO+6 E GENERAZIONE IMPULSO DI STROBE ADCO
ADCOO → MIDEN → SET DI ERROO IN CASO DI ERRORE (PARIO=1 OVVERO PRESENZA PESO 4 NELLA CONFIG. BIT DEL CRT. CON MISSING CLOCK).
ADCOO → EROSN → SET DI ERASO IN CASO DI BLOCCO DELETED (PAROO=0 OVVERO PESO 2 MANCANTE NELLA CONFIGURAZIONE BIT DEL CRT. CON MISSING CLOCK)
1-3

ABILITAZIONE ALLA RICHIESTA ASINCRONA DI µI DI LIV. 1
DEC7 →RESET ADMAO→
DEC7
ADMAN } → SC2N2 →
NREMO
→SCASO→SET DI SCACO
3-6

RICHIESTA ASINCRONA µI DI LIV.1
DEC7 } → SET DI RILIO
SCACO }
6

INVIO STROBE DI SINCRONISMO ECM1

RICHIESTA µI DI LIV.1
ECMIN } → SET DI LIVIO
RILIO } (E RESET DI RILIO)
LIVIO → EPRIN
6-7

CARICO SERIALE DEL CRT. LETTO SU INBIO+PARO+6 E TRASFERIMENTO SU OUTO+7 A FINE CARICO
INBIO } INBIO
CBITN } → PARO+6 → OUTO+7
DEC7O
4

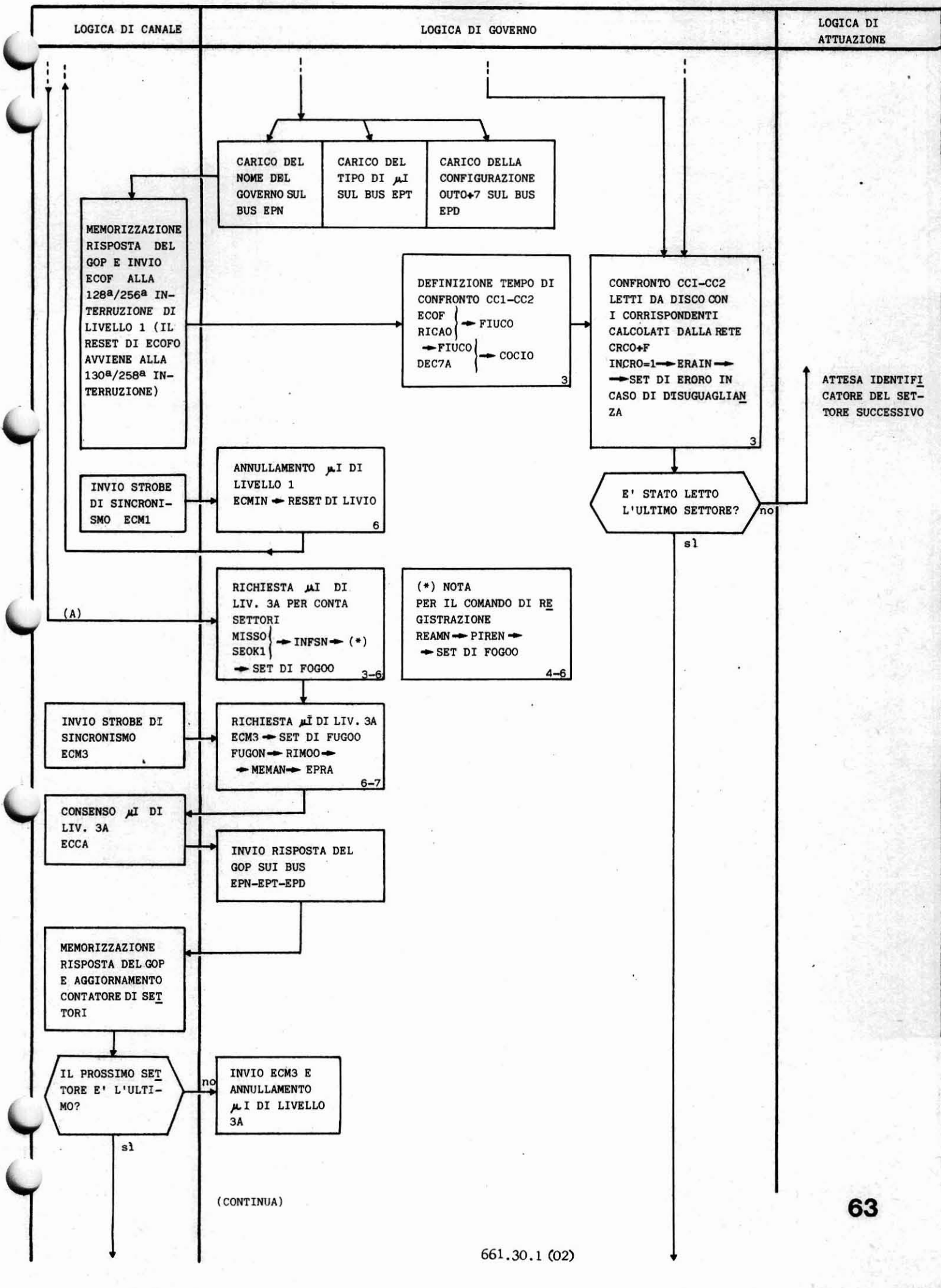
SHIFT SU PARO+7 AD OGNI CBITN E CREAZIONE CODICE CICLICO
CRCF
PAR7N → PAFS } → INCRN
1-4

CONSENSO µI DI LIV. 1 ECC1

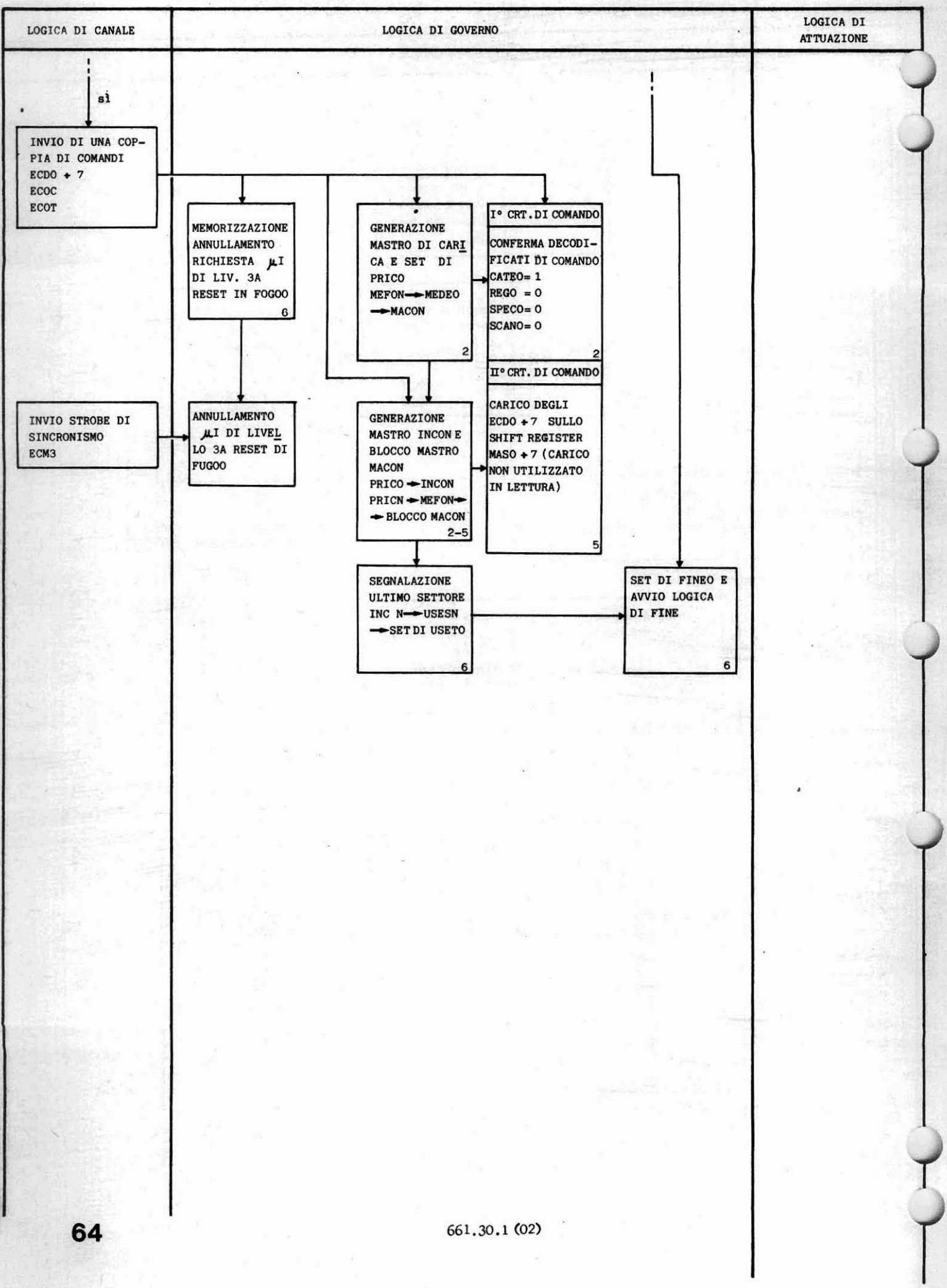
LIVIO →RICAN→ ORMIO=1
RICAO } →EPTO
ERORO }
RICAN=0 → RIRIN=0
6-7

(CONTINUA)

CONTINUAZIONE LETTURA CAMPO DATI



CONTINUAZIONE LETTURA CAMPO DATI



LOGICA DI CANALE	LOGICA DI GOVERNO	LOGICA DI ATTUAZIONE

DALLA FASE DI RICERCA SETTORE

REGOO = 1
SPECO = 0
SCANO = 0
2

RITROVAMENTO SETTORE CERCATO SEOKO = 1
3

BLOCCO DI ORLEO DOPO LA LETTURA DEL 10° CRT DEL GAP
(8) MOD3O → SET
(10) MODON → RESET
} ORROO
ORRON=0 → ORTE →
→ ORLEN → ORLEO = 0
(DALLA FINE DELL'11° CRT. IN POI ORLEO=0 TRAMITE ORREN → ORTE)
2-4-5

ABILITAZIONE ALLA TEMPORIZZAZIONE DI REGISTRAZIONE
ORTEO
MOO4O
} → ORMON →
→ CBITO → DEC7O
3

INIZIO REGISTRAZIONE
ORROO
REGEO
} → ORGEO
→ SET DI ORREO
(ALLA FINE DELL'11° CRT) E REGISTRAZIONE CRT [ØØ] DAL 12° AL 17° COMPRESI
INRON = 1
MOO4N
} → INRIN →
→ INR4O → INREO
4-8

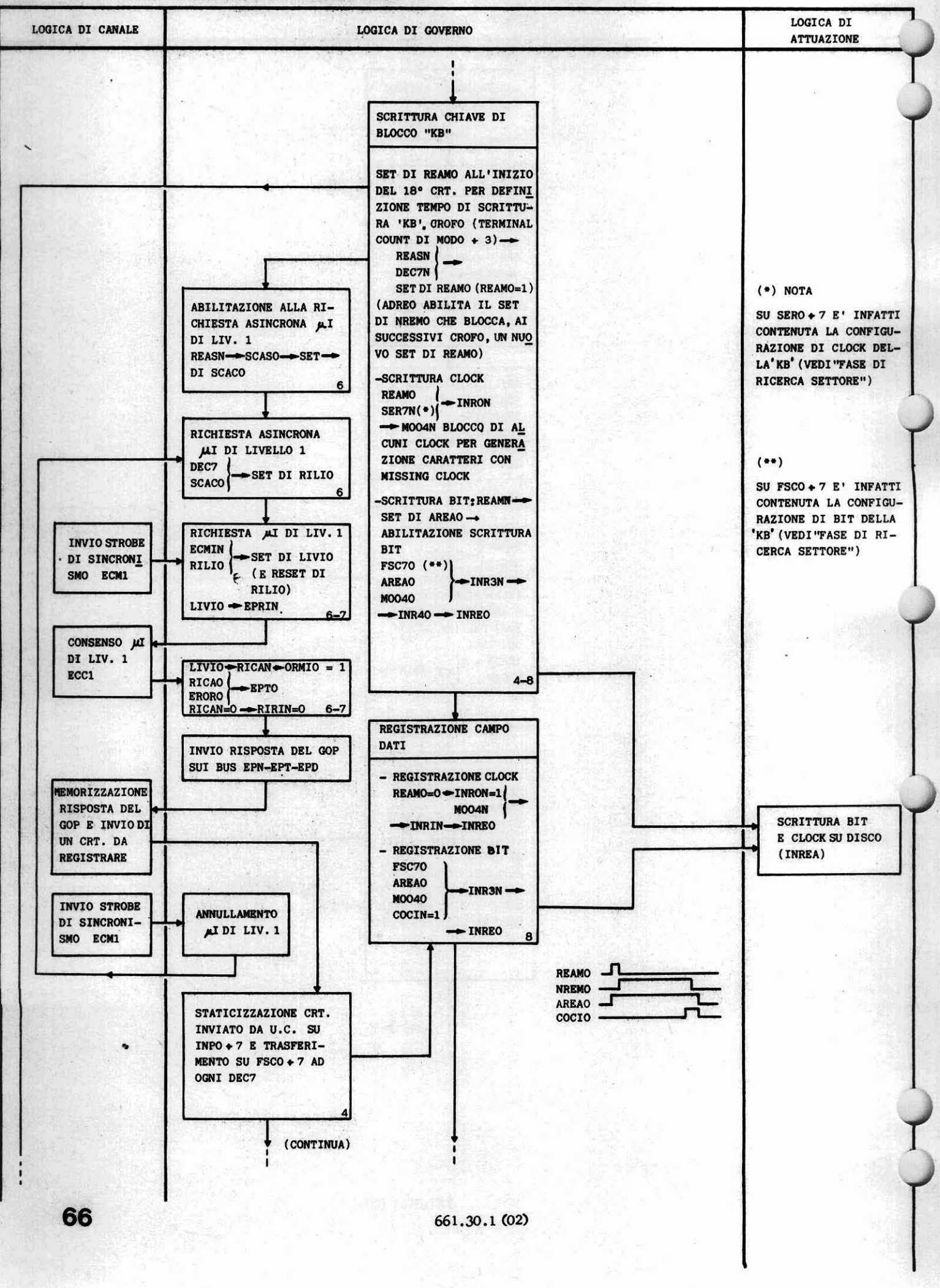
ORREA → ABILITAZIONE CIRCUITO DI REGISTRAZIONE
INREA → INVIO INFORMAZIONI DA REGISTRARE (CLOCK E BIT)

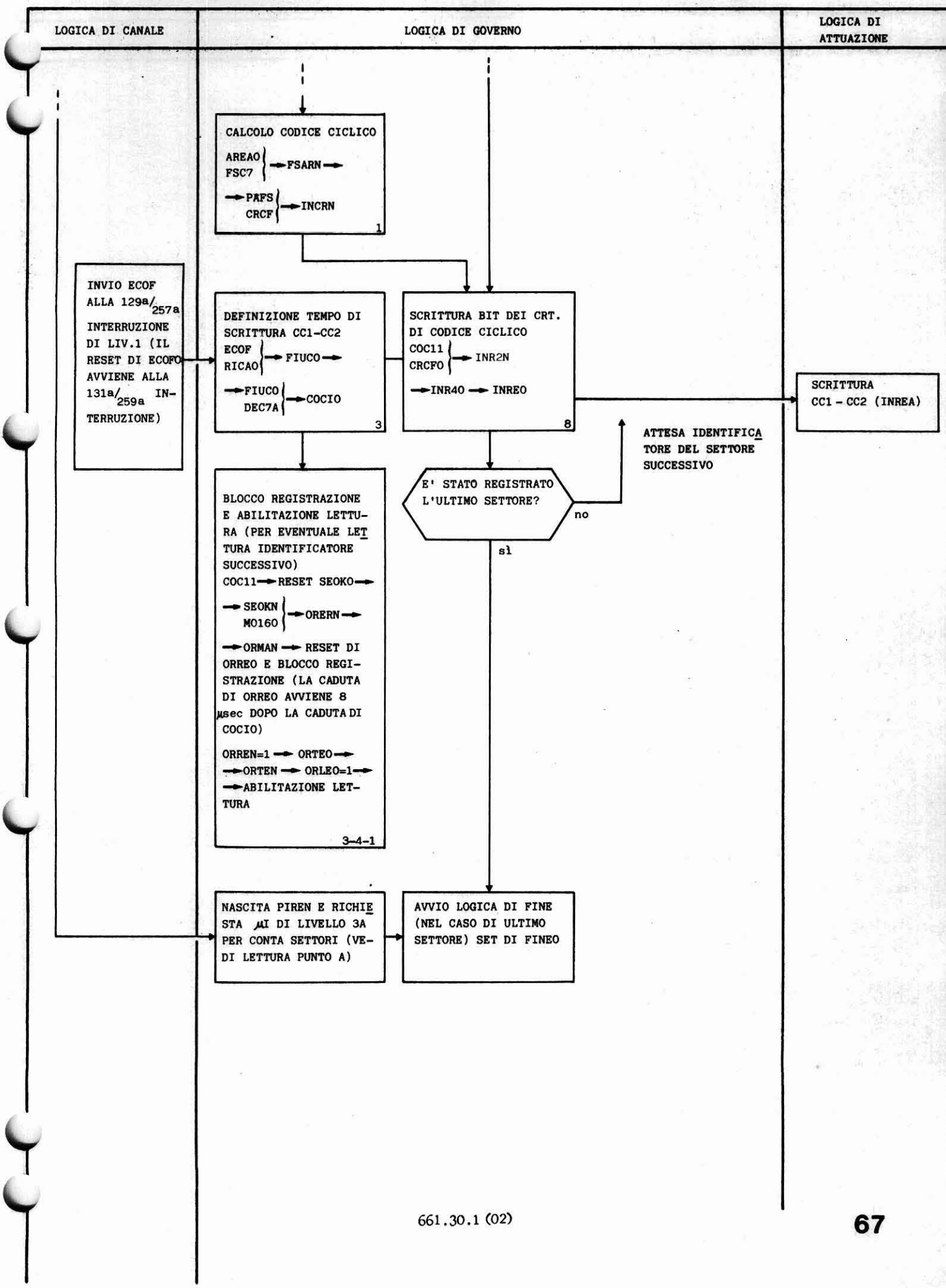
ABILITAZIONE DELLA TESTINA DI "ERASE"
ORNAN
TCO2O
} → SET DI ERGAO
(128 μsec DOPO IL SET DI ORREO)
IL RESET DI ERGAO AVVIENE, NORMALMENTE, CON ONUMO-TCO2O, 464 μSEC DOPO LA CADUTA DI ORREO
4

ERGA → ALIMENTAZIONE DELL'AVVOLGIMENTO DI ERASE

(CONTINUA)

CONTINUAZIONE REGISTRAZIONE DATI




LOGICA DI CANALE
LOGICA DI GOVERNO
LOGICA DI ATTUAZIONE
CALCOLO CODICE CICLICO
AREAO
FSC7
FSARN
PAFS
CRCF
INCRN
1
INVIO ECOF ALLA 129a/257a INTERRUZIONE DI LIV.1 (IL RESET DI ECOFO AVVIENE ALLA 131a/259a INTERRUZIONE)
DEFINIZIONE TEMPO DI SCRITTURA CC1-CC2
ECOF
RICAO
FIUCO
FIUCO
DEC7A
COCIO
3
SCRITTURA BIT DEI CRT. DI CODICE CICLICO
COC11
CRCFO
INR2N
INR40
INREO
8
SCRITTURA CC1 - CC2 (INREA)
ATTESA IDENTIFICATORE DEL SETTORE SUCCESSIVO
BLOCCO REGISTRAZIONE E ABILITAZIONE LETTURA (PER EVENTUALE LETTURA IDENTIFICATORE SUCCESSIVO)
COC11
RESET SEOKO
SEOKN
M0160
ORERN
ORMAN
RESET DI ORREO E BLOCCO REGISTRAZIONE (LA CADUTA DI ORREO AVVIENE 8 µsec DOPO LA CADUTA DI COCIO)
ORREN=1
ORTEO
ORTEN
ORLEO=1
ABILITAZIONE LETTURA
3-4-1
E' STATO REGISTRATO L'ULTIMO SETTORE?
no
sì
NASCITA PIREN E RICHIESTA µI DI LIVELLO 3A PER CONTA SETTORI (VEDI LETTURA PUNTO A)
AVVIO LOGICA DI FINE (NEL CASO DI ULTIMO SETTORE) SET DI FINEO

COMANDO DI VERIFICA

LOGICA DI CANALE	LOGICA DI GOVERNO	LOGICA DI ATTUAZIONE

DALLA FASE DI RICERCA SETTORE

REGOO = 0
SPECO = 1
SCANO = 0 — 2

RITROVAMENTO SETTORE CERCATO
SEOKO = 1 — 3

BLOCCO LETTURA 11° E 12° CRT.
(VEDI LETTURA)

RICONOSCIMENTO CRT. CON MISSING CLOCK (CHIAVE DI BLOCCO)
MISSN → SET DI ADMAO NASCITA DEC7 E SET DI NREMO AL PRIMO CRT. UTILE DEL CAMPO DATI (NREMO RESTA IN SET PER TUTTO IL TEMPO DI LETTURA DEI CRT. DEL CAMPO DATI COMPRESI CC1-CC2)
ADMAN → ABTEO → ABTEE → ABCOO → NASCITA DEC7
ADMAN → ADREO, DEC7 → SET DI NREMO
3-4

STATICIZZAZIONE CRT. CON MISSING CLOCK SU INBIO + PARO + 6 E GENERAZIONE IMPULSO DI STROBE ADCO
ADCOO → MIDEN → SET DI ERROO IN CASO DI ERRORE (PARIO=1 OVVERO PRESENZA PESO 4 NELLA CONFIGURAZIONE BIT DEL CRT. CON MISSING CLOCK).
ADCOO → EROSN → SET DI ERASO IN CASO DI BLOCCO DELETED (PAROO=0 OVVERO PESO 2 MANCANTE NELLA CONFIGURAZIONE BIT DEL CRT. CON MISSING CLOCK)
1-3

ABILITAZIONE ALLA RICHIESTA ASINCRONA μI DI LIV. 1
ADMAO, SPECN → SESCO → SCRON
SET DI SCACO
6

RICHIESTA ASINCRONA μI DI LIVELLO 1
DEC7, SCACO → SET DI RILIO
6

INVIO STROBE DI SINCRONISMO ECM1

RICHIESTA μI DI LIV. 1
ECMIN, RILIO → SET DI LIVIO (E RESET DI RILIO)
LIVIO → EPRIN
6-7

CARICO SERIALE DEL CRT. LETTO SU INBIO + PARO + 6 E TRASFERIMENTO SU OUTO A FINE CARICO
INBIO, CBTIN → INBIO PARO+6 DEC7 → OUTO+7
4

CONSENSO μI DI LIV. 1
ECC1

LIVIO → RICAN → ORMIO=1
RICAO, ERORO → EPTO
RICAN=0 → RIRIN=0
6-7

SHIFT SU PARO + 7 AD OGNI CBITN E CREAZIONE CODICE CICLICO
PAR7N → PAFS, CRCF → INCRN
1-4

INVIO RISPOSTA DEL GOP SUI BUS EPN-EPT-EPD

MEMORIZZAZIONE RISPOSTA DEL GOP E INVIO DI DI UN CRT. DA REGISTRARE (*)

INVIO STROBE DI SINCRONISMO ECM1

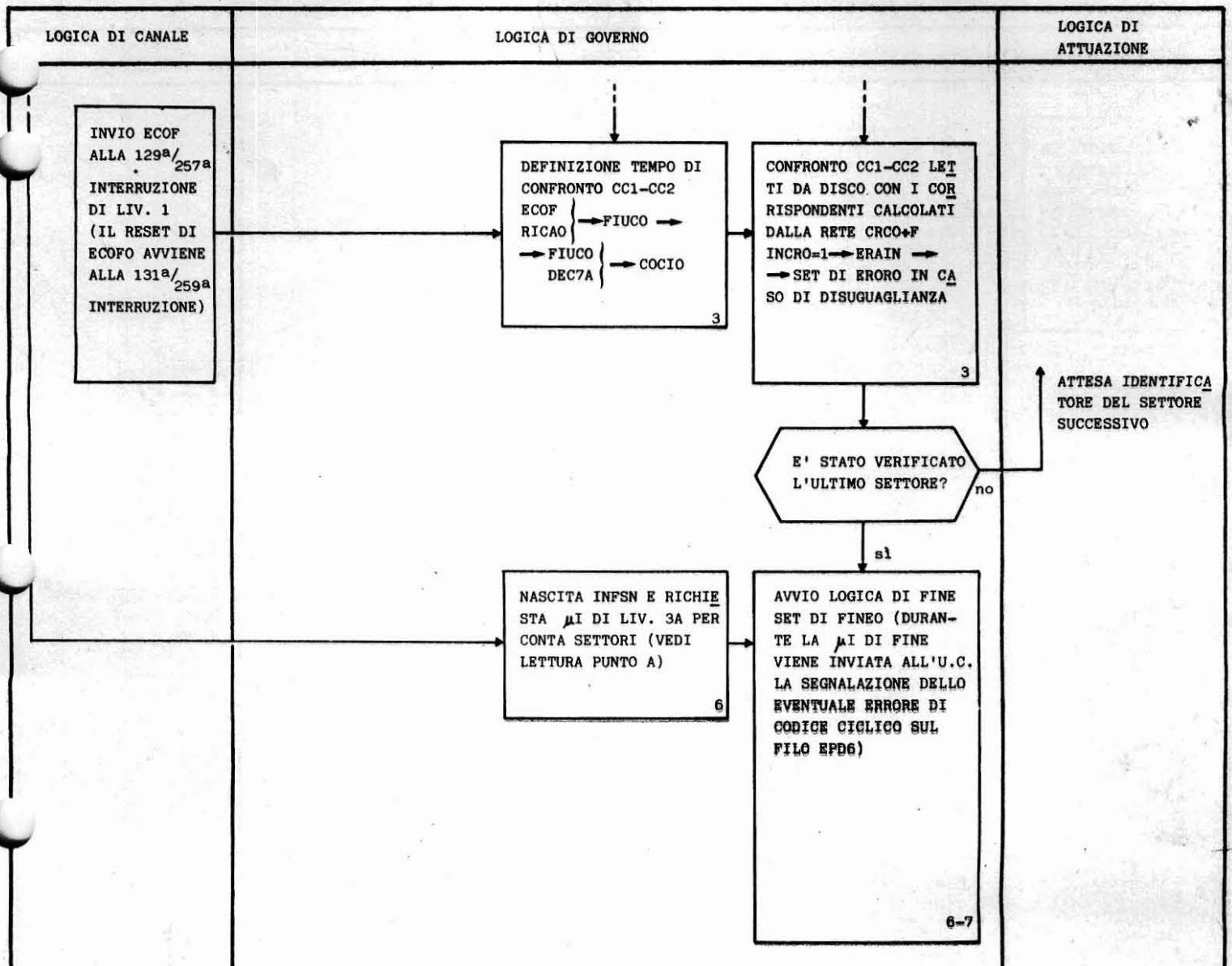
ANNULLAMENTO μI DI LIV. 1

STATICIZZAZIONE CRT. INVIATO DA U.C. SU INPO+7 E TRASFERIMENTO SU FSCO+7 A OGNI DEC7 (NON VIENE REGISTRATO PERCHE' INREN E' BLOCCATO DA ORREO=0)
4-8

(CONTINUA)

(*) NOTA

Infatti il μP usato in verifica è quello di registrazione
(I CRT inviati non sono significativi)


LOGICA DI CANALE
LOGICA DI GOVERNO
LOGICA DI ATTUAZIONE
INVIO ECOF ALLA 129a/257a INTERRUZIONE DI LIV. 1 (IL RESET DI ECOFO AVVIENE ALLA 131a/259a INTERRUZIONE)
DEFINIZIONE TEMPO DI CONFRONTO CC1-CC2
ECOF
RICAO
FIUCO
FIUCO
DEC7A
COCIO
3
CONFRONTO CC1-CC2 LETTI DA DISCO CON I CORRISPONDENTI CALCOLATI DALLA RETE CRCO+F
INCRO=1
ERAIN
SET DI ERORO IN CASO DI DISUGUAGLIANZA
3
ATTESA IDENTIFICATORE DEL SETTORE SUCCESSIVO
E' STATO VERIFICATO L'ULTIMO SETTORE?
no
si
NASCITA INFSN E RICHIESTA μI DI LIV. 3A PER CONTA SETTORI (VEDI LETTURA PUNTO A)
6
AVVIO LOGICA DI FINE SET DI FINEO (DURANTE LA μI DI FINE VIENE INVIATA ALL'U.C. LA SEGNALAZIONE DELLO EVENTUALE ERRORE DI CODICE CICLICO SUL FILO EPD6)
6-7

LOGICA DI CANALE	LOGICA DI GOVERNO	LOGICA DI ATTUAZIONE

DALLA FASE DI RICERCA TRACCIA (SEEK)

REGOO = 0/1
SPECO = 0/1
SCANO = 1

INVIO, DURANTE LA ULTIMA INTERRUZIONE PER CONTA TEMPO, DI UNA COPPIA DI COMANDI
ECDO + 7
ECOC
ECOT

MACON →

I° COMANDO

CONFERMA DECODIFICATI DI COMANDO
REGOO
SPECO
SCANO — 2

INCON →

II° COMANDO

CARICO SU MASO + 7 DEL II° CRT DI COMANDO CONTENENTE LA LUNGHEZZA DELLA CHIAVE DI SCANSIONE (*) — 5

FASE DI RICERCA SETTORE

RICONOSCIMENTO SETTORE CERCATO
SEOKO = 1 — 3

NASCITA INFSN E RICHIESTA μI DI LIVELLO 3A PER CONTA SETTORI (VEDI LETTURA PUNTO A)

BLOCCO LETTURA 11° E 12° CRT. (VEDI LETTURA)

BLOCCO AVVIO REGISTRAZIONE SCANN → REGEO=0 → → ORGEO → BLOCCO SET DI ORREO — 2-4

RICONOSCIMENTO CHIAVE DI BLOCCO: SET DI ADMAO E GENERAZIONE DEC7
MISSN → SET DI ADMAO
ADMAO → ABTEO → ABTEE → ABCOO → NASCITA DEC7 — 3

EVENTUALE SET DI ERROO E ERASO COME IN LETTURA

ABILITAZIONE ALLA RICHIESTA ASINCRONA DI μI DI LIV. 1
ADMAO, SCANN, SESCO → SCRON
→ SET DI SCACO — 6

RICHIESTA ASINCRONA . μI DI LIV. 1
DEC7, SCACO → SET DI RILIO — 6

CARICO LUNGHEZZA CHIAVE DI SCANSIONE SU MODO + 7
MASO + 7, SEOKO, SEADN, DEC7A → MODO + 7 — 5

INVIO STROBE DI SINCRONISMO
ECM1

RICHIESTA μI DI LIV.1
ECMIN, RILIO → SET DI LIVIO (E RESET DI RILIO)
LIVIO → EPRIN — 6-7

CONTA CRT. AD OGNI DEC7 E NASCITA MASTRO DI FINE LETTURA CHIAVE
DEC7 → MODO+7 → MODUO — 5

CONSENSO μI DI LIV. 1
ECC1

LIVIO → RICAN → ORMIO=1
RICAO, ERORO → EPTO
RICAN=0 → RIRIN=0

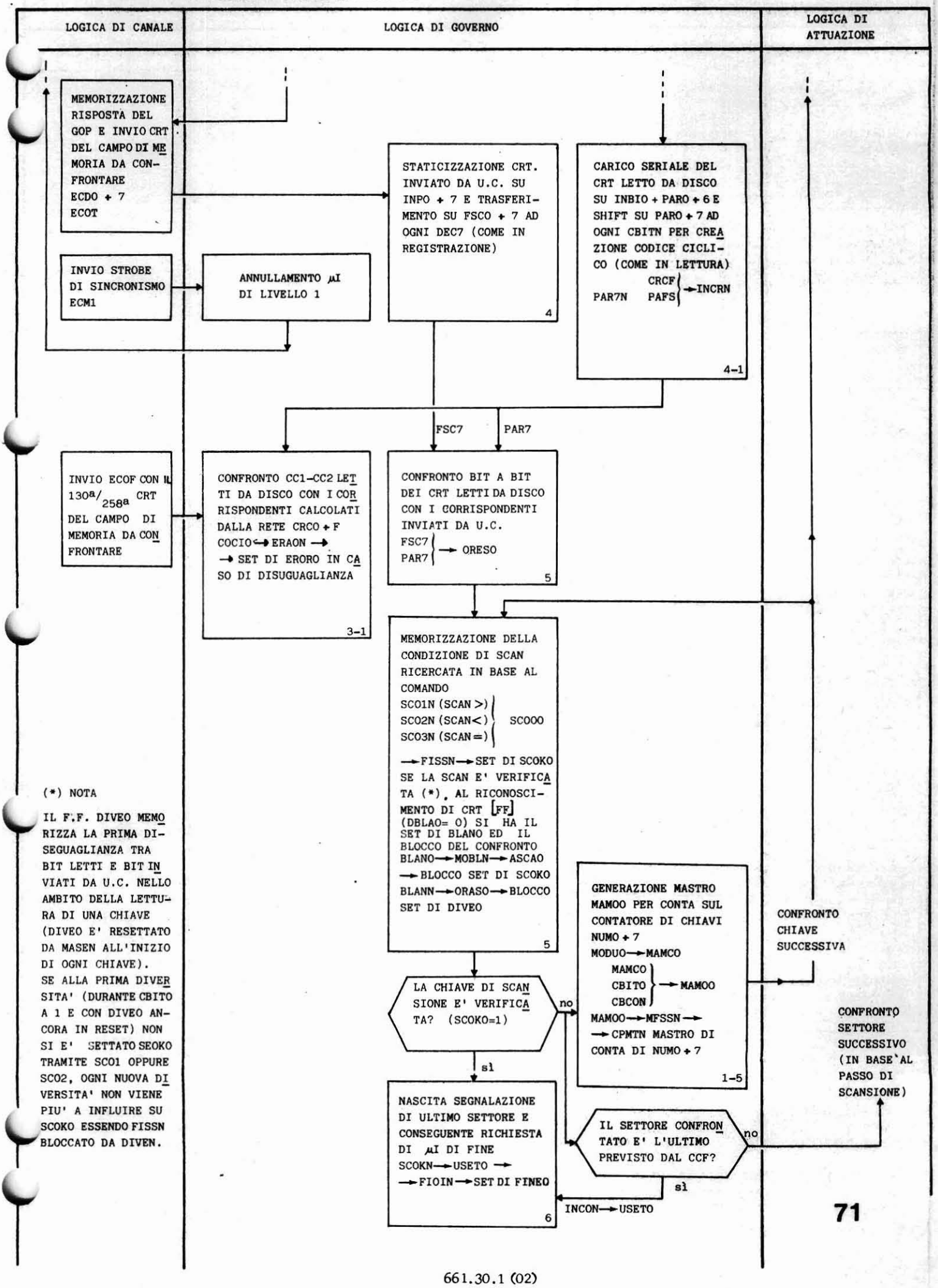
INVIO RISPOSTA DEL GOP SUI BUS EPN-EPT-EPD

(*) NOTA

TALE LUNGHEZZA E' IL COMPLEMENTO A 256 DELLA LUNGHEZZA REALE DELLA CHIAVE DI SCANSIONE.

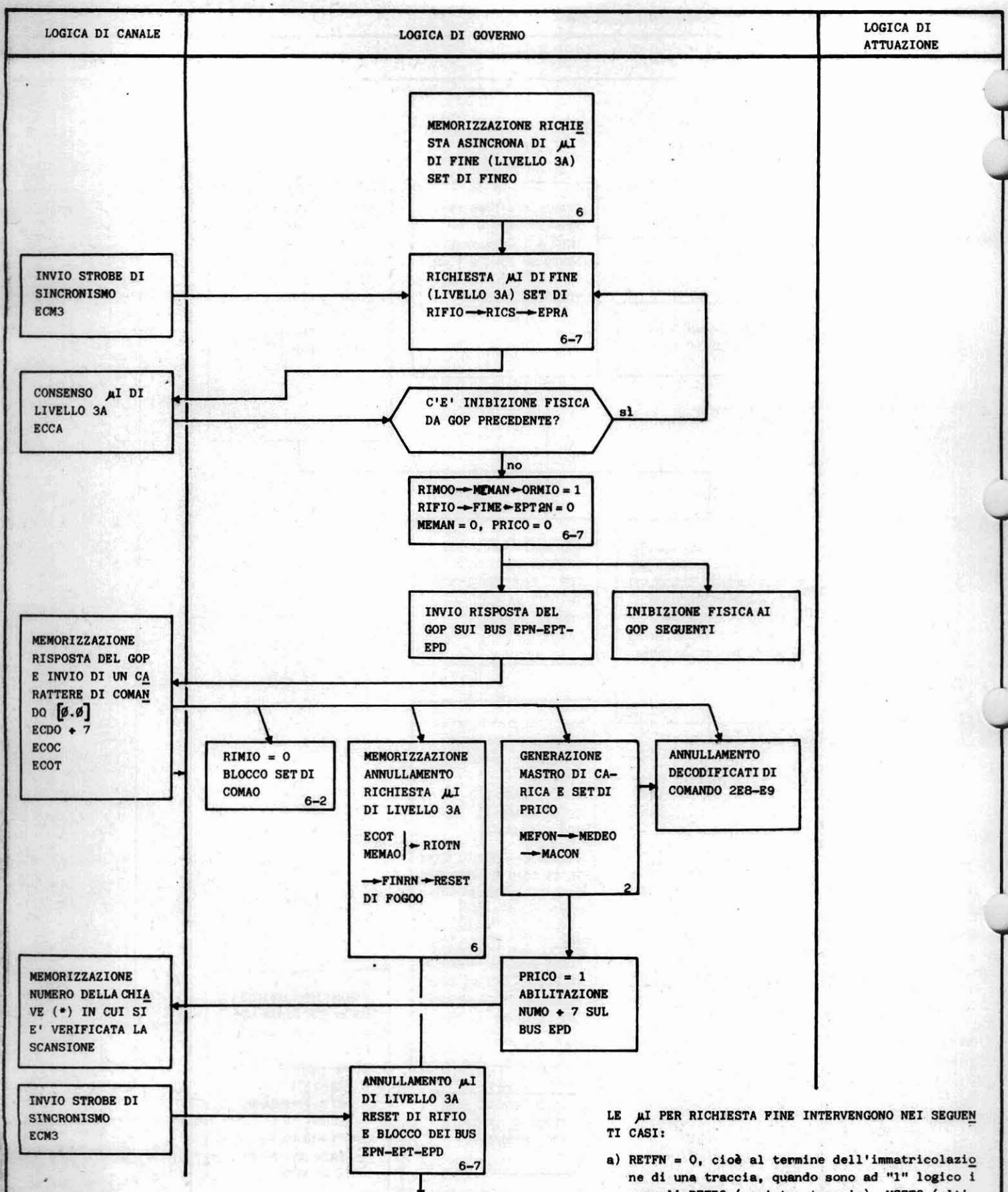
(CONTINUA)

CONTINUAZIONE COMANDO DI SCANSIONE



(*) NOTA

IL F.F. DIVEO MEMORIZZA LA PRIMA DISEGUAGLIANZA TRA BIT LETTI E BIT INVIATI DA U.C. NELLO AMBITO DELLA LETTURA DI UNA CHIAVE (DIVEO E' RESETTATO DA MASEN ALL'INIZIO DI OGNI CHIAVE). SE ALLA PRIMA DIVERSITA' (DURANTE CBITO A 1 E CON DIVEO ANCORA IN RESET) NON SI E' SETTATO SEOKO TRAMITE SCO1 OPPURE SCO2, OGNI NUOVA DIVERSITA' NON VIENE PIU' A INFLUIRE SU SCOKO ESSENDO FISSN BLOCCATO DA DIVEN.



(*) NOTA
SE, AD ES., IL SETTORE E' SUDDIVISO IN 9 CHIAVI, TALE NUMERO AVRA' IL VALORE:
Ø SE LA SCAN E' VERIFICATA SULLA 1ª CHIAVE
2 SE LA SCAN E' VERIFICATA SULLA 3ª CHIAVE
9 SE LA SCAN NON E' VERIFICATA

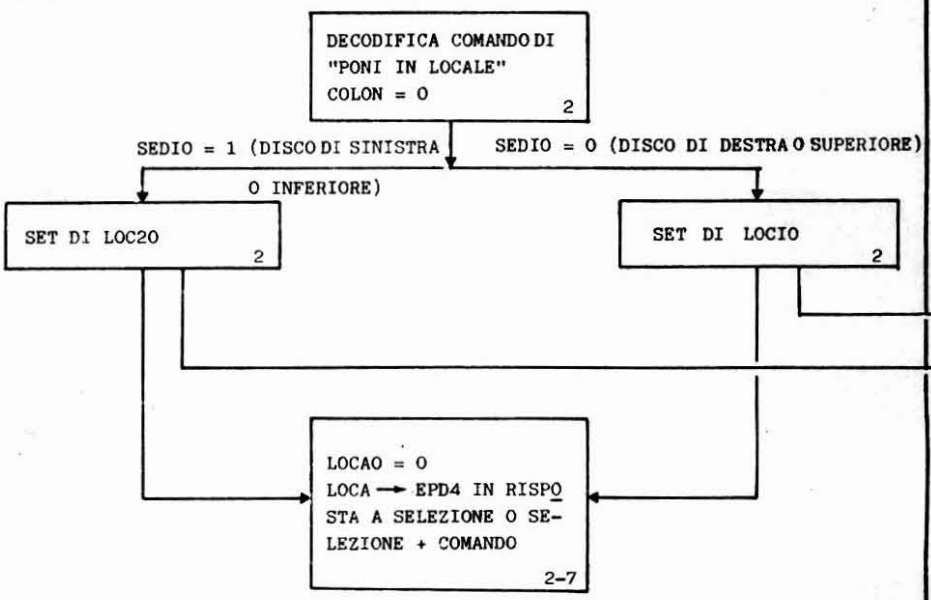
LE μI PER RICHIESTA FINE INTERVENGONO NEI SEGUENTI CASI:

a) RETFN = 0, cioè al termine dell'immatricolazione di una traccia, quando sono ad "1" logico i segnali RETRO (registra traccia), USETO (ultimo settore), INDOO (index).

b) F101N = 0, cioè al termine di una operazione di registrazione o di una operazione di classe C che comporta la lettura di dati dal disco (leggi, verifica, scan.).

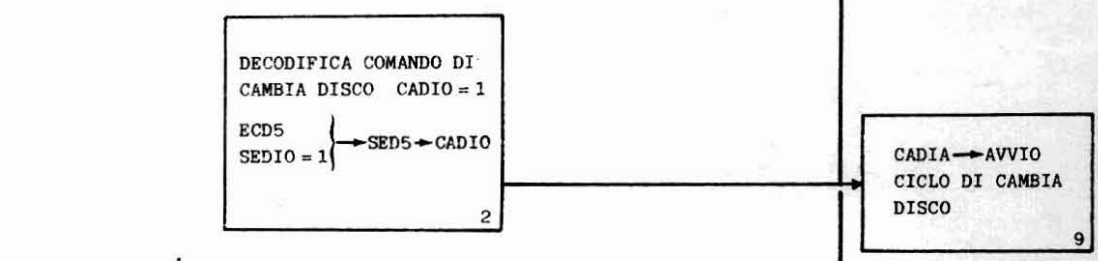
c) F102N = 0, cioè durante la condizione di macchina non disponibile (MAD1B = 1) se è presente il comando di abbassamento testina, oppure al verificarsidell'and tra VIRIO (via ricerca) e SPECN.

COMANDO DI "PONI IN LOCALE"
DALLA FASE DI SELEZIONE + COMANDO



NOTA
IL RESET DELLA CONDIZIONE DI LOCALE AVVIENE ALLA RICHIUSURA DEGLI SPORTELLI (FRONTE DI SALITA DI MAD11) O TRAMITE RESET DA U.C. (ECORB).

COMANDO DI "CAMBIA DISCO"
DALLA FASE DI SELEZIONE + COMANDO



NOTA
IL SEGNALE CADIO NASCE SOLO CON DISCO 2 SELEZIONATO.
IL SEGNALE CICOA INDICA LA CONDIZIONE DI OPERAZIONE DI "CAMBIA DISCO" IN CORSO.